Yann LeCun. The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*.

Jaewoo Lee and Chris Clifton. Differential identifiability. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1041–1049, 2012.

Ilya Mironov. Rényi differential privacy. In *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*, pages 263–275. IEEE, 2017.

Milad Nasr, Reza Shokri, and Amir Houmansadr. Machine learning with membership privacy using adversarial regularization. In *Proceedings of the 2018 ACM SIGSAC conference on computer and communications security*, pages 634–646, 2018.

Milad Nasr, Shuang Songi, Abhradeep Thakurta, Nicolas Papemoti, and Nicholas Carlin. Adversary instantiation: Lower bounds for differentially private machine learning. In *2021 IEEE Symposium on Security and Privacy (SP)*, pages 866–882. IEEE, 2021.

Opacus. Opacus PyTorch library. Available from opacus.ai.

Nicolas Papernot and Thomas Steinke. Hyperparameter tuning with renyi differential privacy. *arXiv preprint arXiv:2110.03620*, 2021.

Nicolas Papernot, Abhradeep Thakurta, Shuang Song, Steve Chien, and Ulfar Erlingsson. Tempered sigmoid activations for deep learning with differential privacy. *arXiv preprint arXiv:2007.14191*, page 10, 2020.

Mark S Pinsker. *Information and information stability of random variables and processes*. Holden-Day, 1964.

Shahbaz Rezaei and Xin Liu. On the difficulty of membership inference attacks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7892–7900, 2021.

Alexandre Sablayrolles, Matthijs Douze, Cordelia Schmid, Yann Ollivier, and Hervé Jégou. White-box vs black-box: Bayes optimal strategies for membership inference. In *International Conference on Machine Learning*, pages 5558–5567. PMLR, 2019.

Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 3–18. IEEE, 2017.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. *Advances in neural information processing systems*, 27, 2014.

Xinyu Tang, Saeed Mahloujifar, Liwei Song, Virat Shejwalkar, Milad Nasr, Amir Houmansadr, and Prateek Mittal. Mitigating membership inference attacks by self-distillation through a novel ensemble architecture. *arXiv preprint arXiv:2110.08324*, 2021.

Anvith Thudi, Ilia Shumailov, Franziska Boenisch, and Nicolas Papernot. Bounding membership inference. *arXiv preprint arXiv:2202.12232*, 2022.

Laurens van der Maaten and Awni Hannun. The trade-offs of private prediction. *arXiv preprint arXiv:2007.05089*, 2020.

Lauren Watson, Chuan Guo, Graham Cormode, and Alexandre Sablayrolles. On the importance of difficulty calibration in membership inference attacks. In *International Conference on Learning Representations*, 2022.

Samuel Yeom, Irene Giacomelli, Matt Fredrikson, and Somesh Jha. Privacy risk in machine learning: Analyzing the connection to overfitting. In *2018 IEEE 31st Computer Security Foundations Symposium (CSF)*, pages 268–282. IEEE, 2018.

A PROOF OF LEMMA 5

Proof. The first part follows by the symmetry of isotropic Gaussian. For the second part (monotonicity) we use the definition of $\mathbf{TV}_a$. Without loss of generality we can assume $a \in [0, 1]$ as otherwise we can work with $\mathbf{TV}_a(P, Q)/a = \mathbf{TV}_{1/a}(Q, P)$. Let $r = \|u_1 - u_2\|_2$. We can show that the derivative of the integral is always positive. In the following calculations, we use c_1, c_2, c_3 and c_4 to denote positive constants that are independent of r.

First note that $x^* = \frac{r^2 - 2\sigma^2 \ln(a)}{2r}$ is a middle point where $e^{-\frac{x^2}{2\sigma^2}} - ae^{-\frac{(x-r)^2}{2\sigma^2}}$ goes from positive to negative as x increases. By our assumption that $a \in [0, 1]$, we have that $x^* > 0$. Recalling that $\mathrm{erf}(z) = \frac{2}{\sqrt{\pi}} \int_0^z \exp(-t^2)dt$, and that $\mathrm{erf}(\infty) = 1$ so that (by symmetry) $\frac{2}{\sqrt{\pi}} \int_{-\infty}^0 \exp(-t^2)dt = 1$, we can write

$$
\begin{aligned}
\mathbf{TV}_a(P, Q) &= c_1 \left(\int_{-\infty}^{\infty} \left| e^{-\frac{x^2}{2\sigma^2}} - ae^{-\frac{(x-r)^2}{2\sigma^2}} \right| dx \right) \\
&= c_1 \left(\int_{-\infty}^{x^*} e^{-\frac{x^2}{2\sigma^2}} - ae^{-\frac{(x-r)^2}{2\sigma^2}} + \int_{x^*}^{\infty} ae^{-\frac{(x-r)^2}{2\sigma^2}} - e^{-\frac{x^2}{2\sigma^2}} \right) \\
&= c_1 \left(1 + \mathrm{erf}\left(x^*/\sqrt{2}\sigma \right) - a\mathrm{erf}((x^* - r)/\sqrt{2}\sigma) \right) \\
&\quad + \left(a(1 - \mathrm{erf}\left((x^* - r)/\sqrt{2}\sigma \right) + (1 - \mathrm{erf}\left(x^*/\sqrt{2}\sigma \right) \right) \\
&= c_2 \left(\mathrm{erf}\left(\frac{r^2 - \ln(a)\sigma^2}{2\sqrt{2}\sigma r} \right) + 1 - a\mathrm{erf}\left(\frac{-r^2 - \ln(a)\sigma^2}{2\sqrt{2}\sigma r} \right) - a \right).
\end{aligned}
$$

Now, let $f_1(r) = \mathrm{erf}\left(\frac{r^2 - \ln(a)\sigma^2}{2\sqrt{2}\sigma r} \right)$ and $f_2(r) = -a\mathrm{erf}\left(\frac{-r^2 - \ln(a)\sigma^2}{2\sqrt{2}\sigma r} \right)$. Taking the derivative with respect to r we have

$$
\begin{aligned}
\frac{\partial f_1}{\partial r} &= c_3 \left(\frac{1}{2\sqrt{2}\sigma} + \frac{\ln(a)\sigma}{2\sqrt{2}r^2} \right) e^{-\left(\frac{r^2 - \ln(a)\sigma^2}{2\sqrt{2}\sigma r} \right)^2} \\
\frac{\partial f_2}{\partial r} &= c_3 a \left(\frac{1}{2\sqrt{2}\sigma} - \frac{\ln(a)\sigma}{2\sqrt{2}r^2} \right) e^{-\left(\frac{-r^2 - \ln(a)\sigma^2}{2\sqrt{2}\sigma r} \right)^2}
\end{aligned}
$$

Now note that we have $e^{-\left(\frac{r^2 - \ln(a)\sigma^2}{2\sqrt{2}\sigma r} \right)^2} = a^{1/2} \cdot e^{-\left(\frac{-r^2 - \ln(a)\sigma^2}{2\sqrt{2}\sigma r} \right)^2}$. Therefore, we have

$$
c_4 \frac{\partial \mathbf{TV}_a}{\partial r} = e^{-\left(\frac{-r^2 - \ln(a)\sigma^2}{2\sqrt{2}\sigma r} \right)^2} \cdot \left(\frac{1 + \sqrt{a}}{2\sqrt{2}\sigma} + \frac{\ln(a)\left(\sqrt{a} - 1\right)\sigma}{2\sqrt{2}r^2} \right).
$$

Now since $a \in [0, 1]$, we have $\ln(a) \leq 0$ and $\sqrt{a} - 1 < 0$, which means the term $\frac{1+\sqrt{a}}{2\sqrt{2}\sigma} + \frac{\ln(a)(\sqrt{a}-1)\sigma}{2\sqrt{2}r^2}$ is positive. This implies that the whole gradient is positive.

□

B ESTIMATING KL WITH RÉNYI DIVERGENCE

For two probability distributions μ and ν, the Rényi divergence of order $\alpha > 1$ is

$$
D_\alpha(\mu \parallel \nu) \triangleq \frac{1}{\alpha - 1} \log \mathbb{E}_{t \sim \nu} \left(\frac{d\mu}{d\nu}(t) \right)^\alpha, \tag{14}
$$

Rényi divergence is non-decreasing in α, and $\lim_{\alpha \to 1} D_\alpha(P \parallel Q) = \mathrm{KL}(P \parallel Q)$.

RDP accounting for DP-SGD. (Abadi et al., 2016; Mironov, 2017) propose methods to account for RDP for the Gaussian mechanism. Implementations of DP-SGD such as Opacus make use of these accounting procedures. This is important as we use these accounting methods to calculate the bound in Equation 14. Specifically, we calculate the D_α for $\alpha = 1 + \tau$ for a very small τ, using the Opacus implementation of Rényi accounting.

C ALGORITHMS

Algorithm 1 Optimal MI

Require: sample rate vector q, number of iterations T, learning rate vector η, noise scale vector σ, gradient norm clip vector r, sample size m

for all $j \in [m]$ **do**
  $\eta_j \leftarrow 0$
for $i \in [T]$ **do**
  for $j \in [m]$ **do**
    sample $t_{i,j} \sim \mathcal{N}(0, \sigma[i]^2 \cdot r[i]^2 \cdot \mathbb{I})$
    $\eta_j \leftarrow \eta_j + \ln\left(1 + q[i](e^{\frac{2t_{i,j}/r[i] - 1}{2\sigma[i]^2}} - 1)\right)$
  $\eta \leftarrow 0$
for $j \in [m]$ **do**
  $\eta_j = \max(\eta_j, 0)$.
  $\eta \leftarrow \eta + \frac{1 - e^{\eta_j}}{m}$
return η

Algorithm 2 DP-SGD (Abadi et al., 2016)

Require: training dataset D, sample rate vector q, number of iterations T, learning rate vector η, noise scale vector σ, gradient norm clip vector r, loss function L

Initiate θ randomly
for $i \in \{T\}$ **do**
  $B_i \leftarrow$ Sample batch via Poisson sampling with rate $q[i]$
  $\nabla[t] \leftarrow \vec{0}$
  for all $(x, y) \in B_t$ **do**
    $\nabla^{(x,y)} \leftarrow$ gradient of $L(x, y)$
    $\overline{\nabla^{(x,y)}} \leftarrow r[t] \cdot \frac{(\nabla^{(x,y)})}{\max(r[t], \|\nabla^{(x,y)})\|_2)}$
    $\nabla[t] \leftarrow \nabla[t] + \overline{\nabla^{(x,y)}}$
  $\widetilde{\nabla[t]} \leftarrow \nabla[t] + \mathcal{N}(0, \sigma[t]^2 r[t]^2 \mathbb{I})$
  $\theta \leftarrow \theta - \eta \widetilde{\nabla[t]}$.
return θ

D PROOF OF THEOREM 18

Proof of Lemma 8. We have

$$\begin{aligned} 2\mathbf{TV}_a(X', Y) = \int_\Omega |d\mu' - a d\nu| &= \int_\Omega |q d\mu - (q + a - 1) d\nu| \\ &= q \int_\Omega \left| d\mu - \frac{(q + a - 1)}{q} d\nu \right| \\ &= 2q\mathbf{TV}_{\frac{a+q-1}{q}}(X, Y). \end{aligned}$$

□

Proof of Theorem 18. The proof steps are similar to Theorem 6. First, we have

$$\begin{aligned} 2\mathbf{TV}(X, Y) = \sum_{s_{\le T-1} \in S_{\le T-1}} &\Pr[X_{\le T-1} = s_{\le T-1}] \cdot \\ &\left(\sum_{s_T} \left| \Pr[X_T = s_T \mid s_{\le T-1}] - \Pr[Y_T = s_T \mid s_{\le T-1}] \frac{\Pr[Y_{\le T-1} = s_{\le T-1}]}{\Pr[X_{\le T-1} = s_{\le T-1}]} \right| \right) \end{aligned}$$

$$= 2 \sum_{s_{\leq T-1} \in S_{\leq T-1}} \Pr[X_{\leq T-1} = s_{\leq T-1}] \mathbf{TV}_{a(s_{\leq T-1})}(X_T \mid s_{\leq T-1}, Y_T \mid s_{\leq T-1}).$$

But since X_T and Y_T are subsampled Gaussian mechanisms we have $X_T \equiv (1-q)Y_T + qX'_T$ where Y and X' are mixtures of Gaussians. Therefore, by Lemma 5 and Lemma 8 we have

$$\begin{aligned}
&\mathbf{TV}(X, Y) \\
&= \sum_{s_{\leq T-1} \in S_{\leq T-1}} \Pr[X_{\leq T-1} = s_{\leq T-1} \in S_{\leq T-1}] q \mathbf{TV}_{\frac{a(s_{\leq T-1})+q-1}{q}}(X'_T \mid s_{\leq T-1}, Y_T \mid s_{\leq T-1}) \text{ (By Lemma 8)} \\
&\leq \sum_{s_{\leq T-1} \in S_{\leq T-1}} \Pr[X_{\leq T-1} = s_{\leq T-1} \in S_{\leq T-1}] q \mathbf{TV}_{\frac{a(s_{\leq T-1})+q-1}{q}}(\mathcal{N}(0, \sigma), \mathcal{N}(r, \sigma)) \text{ (By Lemma 5)} \\
&= \sum_{s_{\leq T-1} \in S_{\leq T-1}} \Pr[X_{\leq T-1} = s_{\leq T-1}] \mathbf{TV}_{a(s_{\leq T-1})}(\mathcal{N}(0, \sigma), (1-q)\mathcal{N}(0, \sigma) + q\mathcal{N}(r, \sigma)) \text{ (By Lemma 8)} \\
&= \sum_{s_{\leq T-1} \in S_{\leq T-1}} \Pr[X_{\leq T-1} = s_{\leq T-1}] \mathbf{TV}_{a(s_{\leq T-1})}(\mathcal{N}(0, \sigma), \mathcal{N}(r \cdot B(q), \sigma)).
\end{aligned}$$

Therefore, we can replace X_T with a mixture of two Gaussians centered at 0 and r and Y_T with a single Gaussian centered at 0. Now we can use the same technique used in proof of Theorem 6 and move X_T and Y_T to the first round and repeat this process. At the end, Y is replaced by a n-dimensional Gaussian centered at 0 and standard deviation σ, and X by a mixture of Gaussians with center randomly selected according to a n-dimensional Bernoulli distribution with probability q. That is, the advantage is bounded by

$$\mathbf{TV}(\mathcal{N}(0^T, \sigma), \mathcal{N}(rB(q)^T, \sigma)).$$

□

E EXPERIMENTAL DETAILS

E.1 GAUSSIAN EXPERIMENT

The simple Gaussian experiment is aimed at stripping away parts of the machine learning pipeline that can interfere with privacy / membership inference, such as the particularities of neural networks or optimization algorithms.

In this setup, $D = \{0, 0, \ldots, 0\}$ and $D' = D \cup \{1\}$. The (clean) summed gradient, before noise addition, is therefore either 0 (on D or on D' if the batch does not contain 1) or 1 (if the batch contains 1). The adversary observes the noisy sums and infers whether they come from D or D'. Given that the adversary knows the distribution is either $\mathcal{N}(0, \sigma^2)$ or $(1-q)\mathcal{N}(0, \sigma^2) + q\mathcal{N}(1, \sigma^2)$, they can perform a simple likelihood test to determine whether the noisy sums come from D or D', and predict the more likely dataset. We report the advantage of this adversary in the "empirical" curve of Figure 6.

F MEMBERSHIP INFERENCE PRECISION

In this section, we refine the analysis of Sablayrolles et al. (2019) for the accuracy of a membership attack.

Upper-bound on precision. Let us first derive a bound on the precision of membership inference. We assume that there are two datasets D and D' and that a differentially-private mechanism $\mathcal{M}$ trains a model represented by θ.

With probability $(1-\delta)$ over the choice of θ, we have:

$$-\epsilon \leq \log\left(\frac{\Pr(M(D) = \theta)}{\Pr(M(D') = \theta)}\right) \leq \epsilon \tag{15}$$

Given that there is a balanced prior $\Pr(D) = \Pr(D')$, using Bayes rule, we have:

$$\Pr(D \mid \theta) = \frac{\Pr(M(D) = \theta)\Pr(D)}{\Pr(M(D) = \theta)\Pr(D) + \Pr(M(D') = \theta)\Pr(D')} \tag{16}$$

$$= \frac{\Pr(M(D) = \theta)}{\Pr(M(D) = \theta) + \Pr(M(D') = \theta)} \tag{17}$$

$$= \sigma\left(\log\left(\frac{\Pr(M(D) = \theta)}{\Pr(M(D') = \theta)}\right)\right), \tag{18}$$

with $\sigma(u) = 1/(1 + \exp(-u))$ the sigmoid function.

Hence the precision $\Pr(D \mid \theta)$ is bounded between $\sigma(-\epsilon)$ and $\sigma(\epsilon)$, as $\sigma(\cdot)$ is non decreasing.

Upper-bound on attack accuracy. The accuracy of the Bayes classifier is

$$\text{Acc} = \max(\Pr(D \mid \theta), 1 - \Pr(D \mid \theta)), \tag{19}$$

and thus

$$\text{Acc} \leq \max(\sigma(\epsilon), \sigma(-\epsilon)) \tag{20}$$

$$= \sigma(\epsilon) \tag{21}$$

This means that the attack accuracy is bounded by $\sigma(\epsilon)$ with probability $1 - \delta$. Empirically, we see that the sigmoid function closely matches the bound given by Humphries et al. (2020). Simply stated, this derivation shows that the bound proven by Humphries et al. (2020) actually holds with probability $1 - \delta$ instead of on average.

$$\begin{aligned}
\text{Acc} &= \frac{1}{2}\left(\Pr(X \in \mathcal{A}) + \Pr(Y \in \mathcal{A}^c)\right) \\
&= \frac{1}{2}\left(\Pr(X \in \mathcal{A}) + 1 - \Pr(Y \in \mathcal{A})\right) \\
&= \frac{1}{2}\left(1 + \text{Adv}\right)
\end{aligned}$$

G Comparing Security Games

In what follows, we write multiple variants of security games specifically defined for DP-SGD. Then we proceed to compare the security games based on an example. We perform experiments on this examples by running an attack and show that our upper bounds are tight even for the weaker security games.

G.1 Our security game with all intermediate gradients (OIG)

1. Adversary picks a datasets $D = \{z_1, \ldots, z_n\}$ and a pair of data points z'_0, z'_1.
2. Challenger samples a bit b uniformly at random and creates

$$D' = \begin{cases} D \cup \{z'_1\} & \text{if } b = 1 \\ D \cup \{z'_0\} & \text{if } b = 0 \end{cases}$$

3. Challenger runs DP-SGD on D' to train a model and sends a transcript of training, including all intermediate gradients, θ (The transcript could only include the final model or more information like the intermediate steps of training) to the adversary.
4. Adversary observes θ and and guesses a bit b'. Adversary wins if $b' = b$.

Remark 10. *In this paper we are interested in analyzing the membership inference advantage for algorithms that could be stated as adaptive composition of sampled Gaussian mechanisms. DP-SGD (Algorithm 2) is a widely used example of such algorithm. Note that we assume that the output of DP-SGD includes all the intermediate gradients that are used to train the model. In other words, the*

parameter θ in the security game contains all the intermediate gradients (not only the final model). However, in what follows we still define the security game for the case that the adversary only sees the final model. We define these more restricted security game so to experimentally compare it with our security game.

Remark 11 (Replacing vs addition/removal)**.** *We also note that in this section we define the notion of advantage for neighboring datasets where one dataset replaces a single example with another example (i.e. the replacement game). The reason for this choice is that the security game of Yeom et. al. is based on replacement and we want to make a fair comparison.*

In the main body, we use the addition/removal definition of neighboring datasets because it is stronger. Namely, we can convert an upper bound on addition/removal security game to an upper bound for the replacement security game.

Remark 12 (Alternative notion of inference in Humphries et al. (2020))**.** *Humphries et al. (2020), propose a new definition for inference attacks. This definition (They call it "Experiment with Data Dependencies".) is is distinct from the previously known notions of membership inference and deals with the ability of an adversary in distinguishing samples from two different distributions. As this notion is a model of distributional inference, the power of adversary in this model is not bounded by differential privacy. The only way to bound this notion of privacy with DP is to pay the cost of group privacy for groups that are almost the same size as of the entire dataset. Hence, we do not study this model in this work.*

G.2 Our security game with final model (OFM)

1. Adversary picks a datasets $D = \{z_1, \dots, z_n\}$ and a pair of data points z'_0, z'_1.
2. Challenger samples a bit b uniformly at random and creates
$$D' = \begin{cases} D \cup \{z'_1\} & \text{if } b = 1 \\ D \cup \{z'_0\} & \text{if } b = 0 \end{cases}$$
3. Challenger runs DP-SGD on D' to train a model and sends the final model θ (The transcript could only include the final model or more information like the intermediate steps of training) to the adversary.
4. Adversary observes θ and and guesses a bit b'. Adversary wins if $b' = b$.

G.3 Yoem et. al's security game with final model (YFM)

1. Challenger samples a dataset $D = \{z_1, \dots, z_{n+1}\}$ from a distribution $\mathcal{D}$.
2. Challenger runs DP-SGD on D to train a model and sends a the final model θ to the adversary.
3. Challenger samples a bit b uniformly at random and creates
$$z' = \begin{cases} z \leftarrow D & \text{if } b = 1 \\ z \leftarrow \mathcal{D} & \text{if } b = 0 \end{cases}$$
4. Adversary observes (θ, z') and and guesses a bit b'. Adversary wins if $b' = b$.

G.4 Yoem et. al's security game with all intermediate gradients (YIG)

1. Challenger samples a dataset $D = \{z_1, \dots, z_{n+1}\}$ from a distribution $\mathcal{D}$.
2. Challenger runs DP-SGD on D to train a model and sends a transcript of training θ, including all intermediate gradients, to the adversary.
3. Challenger samples a bit b uniformly at random and creates
$$z' = \begin{cases} z \leftarrow D & \text{if } b = 1 \\ z \leftarrow \mathcal{D} & \text{if } b = 0 \end{cases}$$
4. Adversary observes (θ, z') and and guesses a bit b'. Adversary wins if $b' = b$.

Notation. For a security model T we use $T(L, n, \mathcal{D})$ to denote the advantage of strongest adversary in that threat model with learning algorithm L, dataset size n and data distribution $\mathcal{D}$ (our threat models do not use this parameter but we include it for symmetry).

Proposition 13. *For any learning algorithm L, any $n \in \mathrm{N}$ and data distribution $\mathcal{D}$ we have* $\mathrm{OFM}(L, n, \mathcal{D}) \leq \mathrm{OIG}(L, n, \mathcal{D})$ *and* $\mathrm{YFM}(L, n, \mathcal{D}) \leq \mathrm{YIG}(L, n, \mathcal{D})$.

Proposition 14 (Proved in Humphries et al. (2020))**.** *For any learning algorithm L, any $n \in \mathrm{N}$ and data distribution $\mathcal{D}$ we have* $\mathrm{YIG}(L, n, \mathcal{D}) \leq \mathrm{OIG}(L, n, \mathcal{D})$ *and* $\mathrm{YFM}(L, n, \mathcal{D}) \leq \mathrm{YIG}(L, n, \mathcal{D})$.

Corollary 15. *Any upper bound on the advantage of adversaries in security game* OIG *is also an upper bound on the advantage of adversaries in the security games* YIG, YFM *and* OFM.

Our analysis above shows that upper bounds for our security model are valid upper bounds for the security game of Yoem et al and Shokri et al as well. Now, to analyze the tightness of our upper bound we perform experiments with attacks in these threat models. We argue that one cannot get a better upper bound on membership inference, unless they make extra assumptions on the data distribution. In what follows, we experimentally verify this.

H Experiments on the optMIAlity of our bound

In this section, we construct a data distribution and study logistic regression on this distribution.

Definition 16. *We define $\mathcal{H}_d$ be the uniform distribution over the hamming ball of radius 1 and centered at zero. Thus, the samples from $\mathcal{H}_d$ have the form, $(0, \ldots, 0, 1, 0, \ldots, 0)$ with only one of the coordinates being 1 and other being 0. For an arbitrary Boolean function $f\colon \{-1, 1\}^d \to \{0, 1\}$ we also define $\mathcal{H}_d^f$ to be the distribution of samples from $\mathcal{H}$ that are labled w.r.t. f, namely, $\mathcal{H}_d^f = (\mathcal{H}_d, f(\mathcal{H}_d))$.*

Experiment setup: We run experiments for membership inference on DP-SGD when trying to learn $\mathcal{H}_d^f$, using logistic regression, and for an arbitrary function f. We set the learning rate to 0.001, the clipping threshold to $.1$ and vary the sub-sampling rate and noise multiplier. We run the models for either 5 or 50 epochs.

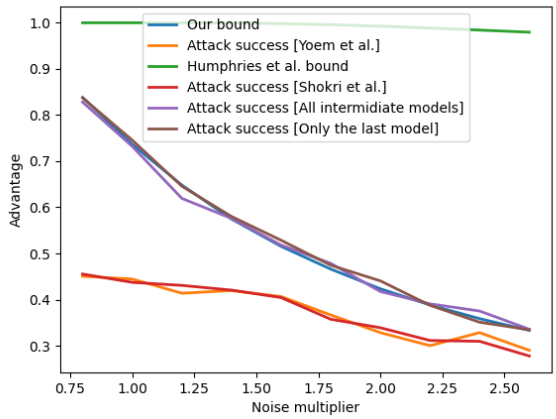
Attacks. We implement a simple attack that only looks at the final model. This adversary looks at the final model θ and the target instance x in hand which is equal to 1 in coordinate i and zero everywhere else. If the ith coordinate of θ is larger than $Tqc/2 + mTqc/d$ (T is the number of iterations, q is sub-sampling rate, n is the number of examples in the training set, and c is the clipping threshold) then the attack predicts $b' = 1$ otherwise it predicts $b' = 0$. We call this attack the "final model attack" (FMA). We also implement another attack that looks at all the intermediate models. This attack basically performs the FMA attack at each iteration and takes majority vote at the end. We call this attack the "Intermediate models attack" (IMA).

Evaluation. We evaluate our FMA attack in the threat models of Yoem et al. and Shokri et al. in the setting where the adversary only sees the final model. We also evaluate this attack in our threat model, in the setting that the adversary only sees the final model. We also report the accuracy of the stronger IMA attack.

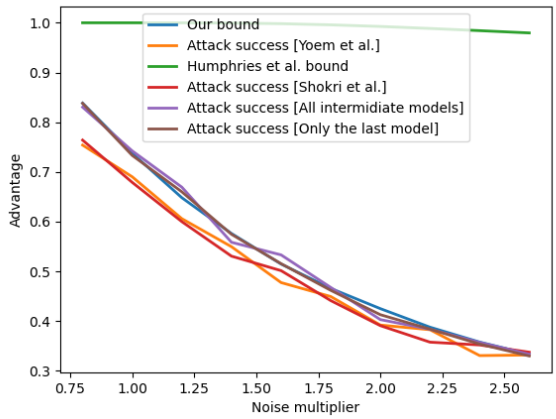
H.1 Results

We now summarize our findings in our experiments

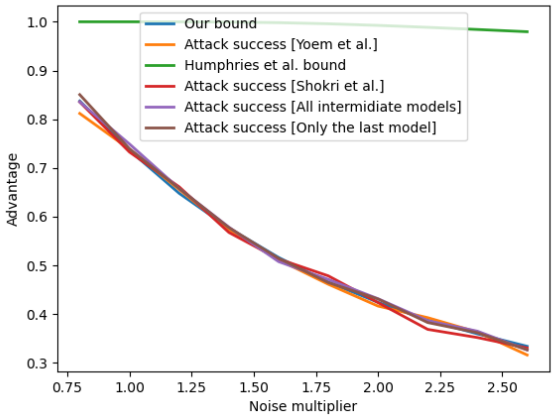
Increasing d reduces the gap between threat models. We instantiate our data distribution $\mathcal{H}_d^f$ with a random function f and select the dimension from $\{2000, 10000, 10000\}$. We set the sample size to 1000, set the sub-sampling rate to $.1$ and vary the noise multiplier to obtain the attack curve. We run the models for 5 epochs and report the attack in various threat models. Our results in Figure H.1 show that by increasing the dimension, the gap between the performance of all attacks and our upper bound shrinks to almost 0. This verifies the optimality of our bound. It also shows that there is no fundamental gap between the threat models and we cannot hope to achieve stronger upper bounds in the weaker threat models, unless we make further assumptions.



(a) $d = 2000$



(b) $d = 10000$



(c) $d = 100000$

Figure 5: Decreasing d results in smaller gap between all threat models and the upper bound. This shows the optimality of our bound in all threat models.

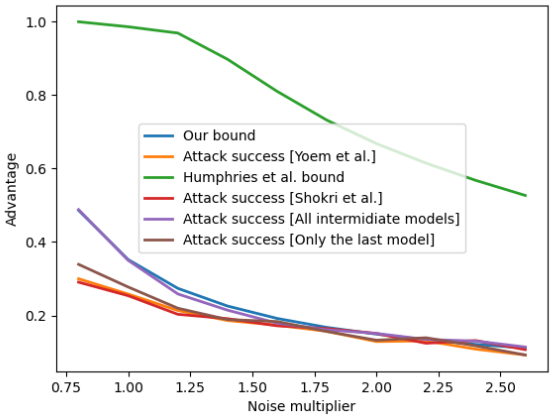


Figure 6: Small sampling rate can produce a gap between FMA and IMA, but this gap tightens as we increase the noise.

Small sub-sampling rate results in a small gap between the performance of FMA **and** IMA**.** We instantiate our data distribution $\mathcal{H}_d^f$ with a random function f and set dimension to 2000 while keeping the number of instances at 1000. We set the sub-sampling rate to .01 and the number of epochs to 50. We vary the noise multiplier to obtain the attack curve. Our results in Figure H.1 show that small sub-sampling rate can create a gap between the performance of two attacks. This gap shrinks as we increase the noise multiplier.

I SECURITY GAME WITH NON-UNIFORM PRIOR

Here, we extend our result to the setting where the prior distribution for the bit b in the security game for membership inference is non-uniform. This setting is recently studied in the work of .

Definition 17 (Non-uniform membership inference.)**.** *We define a security game between an Adversary (who wants to guess training set membership) and a Challenger (who wants to hide training set membership).*

1. *Adversary picks a datasets $D = \{z_1, \ldots, z_n\}$ and a data point z'*
2. *Challenger samples a bit b from a bernouli distribution with probability p and creates*

$$D' = \begin{cases} D \cup \{z'\} & \text{if } b = 1 \\ D & \text{if } b = 0 \end{cases}$$

3. *Challenger runs the a learning algorithm L on D' to train a model and sends a transcript of training θ (The transcript could only include the final model or more information like the intermediate steps of training) to the adversary.*
4. *Adversary observes θ and guesses a bit b'. Adversary wins if $b' = b$.*

We define the advantage of adversary A on learning algorithm L as $\mathbf{Adv}(L, A, p) = 2 \cdot \Pr[b = b'] - 2\max(p, 1-p)$. We also use $\mathbf{Adv}(L, p) = \sup_A \mathbf{Adv}(A, L, p)$ to denote the advantage of any adversary against L.

Note that similar to the uniform setting, with a simple averaging argument we can show that the best adversarial strategy in the non-uniform membership security game is a deterministic strategy. Therefore, assuming $p < 0.5$, the advantage for the learning algorithm L is then defined as

$$\frac{\mathbf{Adv}(L, p)}{2} = \sup_{\mathcal{A}} \mu(\mathcal{A}) \cdot p + (1 - \nu(\mathcal{A})) \cdot (1-p) - (1-p) \tag{22}$$

$$= \sup_{\mathcal{A}} (1-p) \left(\nu(A) - \frac{p}{1-p} \mu(A) \right) \tag{23}$$

$$= \sup_{\mathcal{A}} (1-p) \left(\nu(A) - \frac{p}{1-p} \mu(A) \right) - (1-p) \left(\nu(\bar{A}) - \frac{p}{1-p} \mu(\bar{A}) \right) \tag{24}$$

$$+ (1-p) \left(\nu(\bar{A}) - \frac{p}{1-p} \mu(\bar{A}) \right) \tag{25}$$

$$= 2(1-p)\mathbf{TV}_{\frac{p}{1-p}}(X, Y) + (1 - 2p - \frac{\mathbf{Adv}(L,p)}{2}). \tag{26}$$

Therefore we have

$$\mathbf{Adv}(L,p) = 2(1-p)\mathbf{TV}_{\frac{p}{1-p}}(X, Y) + 1 - 2p \tag{27}$$

Now using this, we can prove the following Theorem.

Theorem 18 (Non-uniform gaussian Composition with sub-sampling)**.** *Let $M_1, \ldots, M_T$ be a series of adaptive Gaussian Mechanisms with L_2 sensitivity r and Gaussian noise with standard deviation σ and sub-sampling rate q. The non-uniform membership inference risk of the composition of M_i's is at most*

$$2(1-p)\mathbf{TV}_{\frac{p}{1-p}}\left(\mathcal{N}(0^T, \sigma), \mathcal{N}(r \cdot B(q)^T, \sigma)\right) + 1 - 2p$$

where $p < 0.5$ is the probability of sampling of the additional example in the non-uniform security game.

Proof. The proof is similar to the proof of Theorem 6 except that we use Equation 27 instead of Equation 4. □