Differential Privacy Defenses and Sampling Attacks for Membership Inference

Shadi Rahimian[1] **Tribhuvanesh Orekondy**[2] **Mario Fritz**[1]
[1] CISPA Helmholtz Center for Information Security, Germany
[2] Max Planck Institute for Informatics, Germany
{shadi.rahimian, fritz}@cispa.saarland orekondy@mpi-inf.mpg.de

Abstract

Machine learning models are commonly trained on sensitive and personal data such as pictures, medical records, financial records, etc. A serious breach of the privacy of this training set occurs when an adversary is able to decide whether or not a specific data point in her possession was used to train a model.

We protect the model by adding noise to the gradients and as an alternative method, add noise only to the logits to protect the output of the model and evaluate the effect of these two differentially-private techniques against membership inference attacks on 5 diverse datasets. While all previous membership inference attacks rely on access to the posterior probabilities, we present the first attack which only relies on the predicted class label - yet shows high success rate.

1 Introduction

Studies inferring presence of certain individuals from summary statistics of a population dates back to [6]. More recently, Shokri et al. [10] demonstrated membership inference can be similarly performed on black-box machine learning (ML) models. Such attacks are especially severe given the proliferation of ML models (e.g., cloud service APIs, medical diagnosis) and the privacy sensitivity of the training data used to train these models.

There are serious risks associated with membership inference attacks and it is important to protect the learning models and the individuals whose data was used to train these models against the membership inference adversary. A general framework that is commonly used for privacy is *differential privacy* [4, 2, 3] (DP), which offers rigorous mathematical guarantees of the privacy of the individuals whose data is contained in a database. Differential privacy relies on methodical perturbation of the algorithm that is applied on a database such that the presence or the absence of an individual's data in that database is not observable by any adversary. This perturbation is usually done via adding noise and the privacy budget or spending, ϵ, is inversely proportional to the amount of noise. However, there seems to be no systematic study on the success of defending membership inference attacks with DP. Previous studies connecting these two [8, 7] rely only on the gradient perturbation and are mainly focused on the relationship between the privacy budget and the degraded utility of the model, rather than the performance of the attacker.

Contributions. In this paper, we try to have a deeper, more comprehensive look at the membership inference attacks and DP methods to defend against these attacks. Our main contributions are:

a. We study the effect of applying DP-SGD [1] to our models on the membership inference attacks and unlike the original attack algorithms in [9], we do not restrain ourselves by choosing thresholds to measure the success rate of the adversary and instead report the AUC values. We carry out our experiments using MNIST, FashionMNIST, CIFAR10, CIFAR100 and Purchase100 datasets.

33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

b. As an alternative to DP-SGD, we propose a faster method which protects the output of the black-box models. We achieve this by adding noise to the logits only at the querying time of a trained model and study the performance of the membership inference attacker.
c. All the membership inference attacks that we are aware of use the posterior information from the victim model. We suggest a novel attack model which can work only with the argmax of the posterior vector from the model. In this method, the attacker depends on generating multiple noisy samples from each data point. We refer to this as *sampling attack*.
d. We mitigate the success of the sampling attack with a randomized response algorithm [12, 5] that flips the returned class labels.

2 Method and Experiments

2.1 Attack Technique

Central to performing the membership inference attack of Shokri et al. [10] is training multiple shadow models (which mimics the black-box behaviour of the victim ML model) and attack models (binary membership classifiers). Consequently, the approach depends heavily on an attacker with access to the same training data distribution as that of the victim model. To circumvent the attacker's access to training data and additionally operating under weaker assumptions, Salem et al. [9] demonstrated that effective membership inference are possible. We choose the most versatile adversarial model of [9] to inspect membership inference attacks on our dataset:

LRN-Free Adversary. This adversarial model requires no shadow model or access to data from the same distribution as the training set of the victim model. At attack time, the adversary queries the victim model by the data point under attack and directly inspects the posterior vector. If the maximum value of the posterior vector is above a certain threshold, it is hypothesized that the point belonged to the training set of the model therefore the model is more confident about it and thus that data point is classified as *in* (member of training set), otherwise as *out*. However, different from the original paper, we refrain from choosing thresholds. Instead we calculate the Area Under the ROC Curve (AUC) for all the possible thresholds. Since, as opposed to other adversarial techniques, this procedure requires no training of any shadow or attack model, we refer to this method as learning-free adversary.

2.2 Defenses

DP-SGD. The first defense mechanism that we study is DP-SGD [1], which adds noise to the l_2-clipped gradients during the Stochastic Gradient Descent (SGD) step of training. The task of calculating the accumulated privacy budget over the course of training is done by *moments accountant* [1].

DP-Logits. A drawback of the DP-SGD method is that due to adding noise at each epoch of training, the accumulated value of ϵ grows to very large numbers. It also slows down the process of training. Combined with the fact that we assume the adversary would not have access to the internal parameters of the network, we decided to use a method that protects only the output. For this, we train the models with no DP method but use the Gaussian Mechanism to add noise to l_2 normalized logits when the network is queried:

$$\mathbf{l}(x_i) \leftarrow \mathbf{l}(x_i)/\max(1, \frac{\|\mathbf{l}(x_i)\|}{S}), \qquad \text{for clipping norm } S$$

where $\mathbf{l}(x_i)$ indicates the vector of logits for the input x_i. We call this method DP-Logits for short. The privacy budget for this method and for the (ϵ, δ) differentially-private Gaussian Mechanism with the noise scale σ and l_2 sensitivity S can be calculated as: $\sigma \geq \frac{S}{\epsilon}\sqrt{2\ln(\frac{1.25}{\delta})}$ for $\delta \propto \frac{1}{|d|}$ where $|d|$ is the size of the training dataset.

2.3 Sampling Attack

In general, all the membership inference adversarial techniques depend on the information obtained from the posterior vectors of the victim model. This tempts us to suggest that by avoiding to return the posterior vector, and just reporting the most confident 'argmax' label $k = \arg\max_k P(y = k|x_i)$

of it, we can defeat the adversary, completely. This is in fact true for all the previously-suggested membership inference attack methods, however, we propose a new technique which can help the adversary reconstruct the posterior vector. In this method, the adversary generates n noisy samples from each data point and formulates the posterior vector as

$$\hat{p}(y = k|x_i) = \frac{1}{n}\sum_{i=1}^{n_k} 1, \qquad \text{for } n \text{ noisy samples generated from } x_i$$

where n_k is the number of samples that are returned with label k. She can then use the maximum value of this reconstructed posterior, $\hat{p}$, and utilize LRN-Free adversary to attack. For this method, choosing the best noise level is crucial for the success of the attacker. The attacker with access to data from the same distribution as the training set of the victim model can query a trained shadow model with noisy samples for different values of noise and choose the optimal noise level corresponding to the most successful attack. This means that for this attack method an access to the dataset and a shadow model is required, however, no attack model is necessary.

2.4 Defense Against the Sampling Attack

As a defense against the sampling attack, we propose using the randomized response [12, 5] mechanism with a fair coin. Since we have more than 2 classes, we flip the coin twice and with a total probability of 0.75% keep the returned label otherwise uniformly at random choose among the other labels. This means that the privacy budget is

$$\frac{1}{n}\epsilon = \ln\left(\frac{\Pr[\text{returned label} = k|\text{true label} = k]}{\Pr[\text{returned label} = k|\text{true label} = k']}\right) = \ln\left(\frac{0.75}{0.25/(N_c - 1)}\right) = \ln 3(N_c - 1)$$

where N_c indicates the total number of classes in the dataset and the factor $\frac{1}{n}$ is the graceful deterioration of privacy due to querying the same point n times.

3 Experimental Setup

Since our focus is on Machine Learning as a Service (MLaaS), whenever the adversary requires a shadow model, such as in the sampling attack technique, we assume that she has access to the same model as the victim. Thus the structure of the neural network and the defense mechanisms applied to the network (e.g. DP-SGD) are similar between the victim model and the shadow model. We used a densely-connected network with 3 hidden layers for Purchase100 and a VGG [11]-inspired convolutional neural network for other datasets.

For each dataset, we combine the training and the test sets and divide this collection into 4 equal parts. One part is used to train the shadow model (*in*), one part as the test set (*out*) of the shadow model, and the other two parts are used in the same way for the victim model.

MNIST	98
FashionMNIST	88
CIFAR10	69
CIFAR100	35
Purchase100	80

Table 1: Accuracies of undefended victim models.

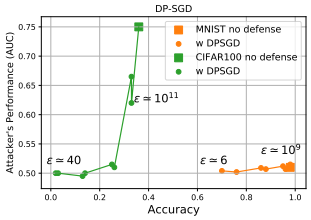
Sampling attack. For the sampling attack, for image datasets we calibrate the values in each channel to be in range $[0, 1.0]$. Then we add noise from a Gaussian distribution with $\rho = \{i \times 0.05 | 0 \leq i < 20,\ i \in \mathbb{N}\}$ to each pixel of the data in each channel, independently.

For Purchase100 which consists of binary features, we randomly flip the value of each feature from 0 to 1 or vice versa, with an increasing probability of flipping $\rho = \{i \times 0.05 | 0 \leq i < 20,\ i \in \mathbb{N}\}$.

We generate $n = 100$ samples from each data point.

4 Results

DP-SGD and DP-Logits. To demonstrate the effect of applying DP on membership inference attacks, we choose to plot AUC values versus accuracy of the model for different levels of noise multiplier, m, which is defined as $m = \sigma/S$, where σ is the standard deviation of the Gaussian distribution that the noise (for the DP mechanisms) is drawn from and S is the l_2 norm clipping threshold. Note that AUC $= 0.5$ means the attack is at chance level and completely unsuccessful and



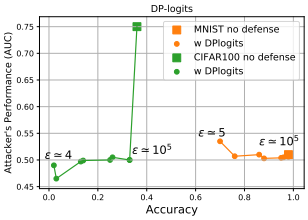


Figure 1: Performance of the attacker versus the accuracy for DP-SGD and DP-Logits. Different noise multiplier levels of DP algorithms are connected with the line in an increasing order.

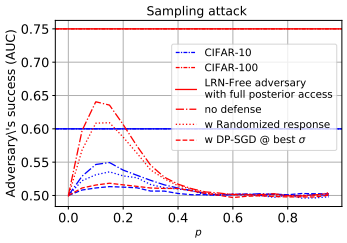


Figure 2: Sampling attack vs. Randomized Response and DPSGD

AUC $= 1.0$ indicates that the attacker is 100% successful in the membership inference attack. Fig. 1 shows the effect of applying DP-SGD and DP-Logits on the performance of the attacker for different noise levels. Here we choose to include two datasets that contrast the most in the accuracy of the model with no DP method applied (as stated in Table. 1). We find:

(i) Models that generalize better and have higher accuracy, e.g. MNIST and FashionMNIST, are less prone to the membership inference attacks.
(ii) Compared to DP-SGD, the performance of the attacker drops faster at lower noise levels for the DP-Logits method.

Note that the privacy budget has inverse relationship to the noise level. So from a DP point of view, we are searching for a point with the highest noise level that still has a good utility compared to the base classifier. The optimal values of the noise multiplier for the 3 most interesting datasets, where the attacker is initially successful, are listed in Table. 2. The corresponding ϵ values for these optimal noise levels are also listed in the same table. (See Appendix for a more comprehensive overview of the results on each dataset)

Method	CIFAR10	CIFAR100	Purchase100
m^*_{DPSGD}	0.001	0.005	0.01
ϵ_{DPSGD}	$\sim 10^9$	$\sim 5 \times 10^7$	$\sim 10^7$
$m^*_{\text{DP-Logits}}$	0.01	0.001	0.001
$\epsilon_{\text{DP-Logits}}$	~ 500	~ 5000	~ 5000

Table 2: Optimal values of noise multipliers and corresponding privacy budget, for both methods.

Sampling attack vs randomized response and DP-SGD. In Fig. 2 we demonstrate the results for sampling attack on two datasets which are most prone to membership inference attacks and show how the randomized response or DPSGD-trained models can mitigate the risks of sampling attack used as a membership inference technique. We observe that with sampling attack, the adversary is able to retain almost half of her performance at the optimal input perturbation level (ρ) compared to when the full posteriors are accessible. We also observe that DP-SGD trained models work better against the sampling attack compared to when randomized response mechanism is used as a defense.

5 Conclusion

The lack of a systematic study on the effect of DP for defending against membership inference attacks motivated us for this paper. We observed that, aside from DP-SGD that perturbs and protects the whole model, we can use an output perturbation mechanism such as Gaussian Mechanism to achieve satisfying results against the membership inference attackers with lower privacy spending and faster implementation and computation. Our novel sampling attack method, does not depend on the posterior vectors and the attacker can gain a significant portion of her initial performance just by knowing the class label of the data points. These results might improve further if we generate $n > 100$ noisy samples. The DP-SGD as well as the randomized response mechanism can protect the model against the sampling attack. In this work, we only study the effect of using a fair coin with a total probability of 75% to return the true answer (after being flipped twice). We expect that by using an unfair coin and reducing the probability of returning the true answer, the attacker's performance would drop even further. Nevertheless, models that have been trained with DP-SGD also offer acceptable protection against sampling attacks.

References

[1] Martin Abadi et al. “Deep learning with differential privacy”. In: *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM. 2016, pp. 308–318.

[2] Cynthia Dwork. “A firm foundation for private data analysis”. In: *Communications of the ACM* 54.1 (2011), pp. 86–95.

[3] Cynthia Dwork et al. “Calibrating noise to sensitivity in private data analysis”. In: *Theory of cryptography conference*. Springer. 2006, pp. 265–284.

[4] Cynthia Dwork et al. “Robust traceability from trace amounts”. In: *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*. IEEE. 2015, pp. 650–669.

[5] Bernard G. Greenberg et al. “The Unrelated Question Randomized Response Model: Theoretical Framework”. In: *Journal of the American Statistical Association* 64.326 (1969), pp. 520–539. DOI: 10.1080/01621459.1969.10500991.

[6] Nils Homer et al. “Resolving individuals contributing trace amounts of DNA to highly complex mixtures using high-density SNP genotyping microarrays”. In: *PLoS genetics* 4.8 (2008), e1000167.

[7] Bargav Jayaraman and David Evans. “Evaluating Differentially Private Machine Learning in Practice”. In: *28th USENIX Security Symposium (USENIX Security 19). Santa Clara, CA: USENIX Association*. 2019.

[8] Md Atiqur Rahman et al. “Membership Inference Attack against Differentially Private Deep Learning Model.” In: *Transactions on Data Privacy* 11.1 (2018), pp. 61–79.

[9] Ahmed Salem et al. “ML-Leaks: Model and Data Independent Membership Inference Attacks and Defenses on Machine Learning Models”. In: *Annual Network and Distributed System Security Symposium (NDSS)*. to appear. Feb. 24, 2019. published.

[10] Reza Shokri et al. “Membership inference attacks against machine learning models”. In: *2017 IEEE Symposium on Security and Privacy (SP)*. IEEE. 2017, pp. 3–18.

[11] Karen Simonyan and Andrew Zisserman. “Very deep convolutional networks for large-scale image recognition”. In: *arXiv preprint arXiv:1409.1556* (2014).

[12] Stanley L. Warner. “Randomized Response: A Survey Technique for Eliminating Evasive Answer Bias”. In: *Journal of the American Statistical Association* 60.309 (1965). PMID: 12261830, pp. 63–69. DOI: 10.1080/01621459.1965.10480775.

method	optimizer	learning rate	batch size
DP-SGD	AdamOptimizer	0.001	128
DP-Logits	AdamOptimizer	0.001	128

Table 3: The parameters used for DP-SGD and DP-Logits methods.

FashionMNIST	MNIST	CIFAR10	CIFAR100	Purchase100
22	20	14	63	478

Table 4: The l_2 norm values at which the logits were clipped

A Randomized Response

For the randomized response mechanism we can calculate the expected accuracy (utility) of the model as follows

$$\begin{aligned} \text{accuracy} &= \frac{n_T}{n_T + n_F} \\ \rightarrow \mathbb{E}[\text{accuracy}_{DP}] &= \frac{0.75 * n_T}{n_T + n_F} + \frac{0.25/(N_c - 1) * n_F}{n_T + n_F} \end{aligned}$$

where n_T and n_F indicate the number of points which correctly match the ground truth labels and the number of points which deviate from the ground truth labels, respectively.

B Parameters

All our experiments were implemented in Tensorflow. The parameters used for each DP method is listed in Table. 3. The most important parameters for both DP-SGD and DP-Logits methods are the l_2 clipping threshold, S, which bounds the sensitivity of the function and the standard deviation of the Gaussian noise, σ. The effect of these two parameters is intertwined and larger values of noise are needed for larger clipping thresholds. In general, the clipping threshold should be set to a percentile of the parameter that we want to clip.

For DPSGD, we set $S = 3$ for CIFAR10 and CIFAR100 and $S = 1$ for the rest of data sets. For DP-Logits after inspection of the histograms of the l_2 norms of the logits we chose the values shown in the Table. 4.

We report our results in terms of noise multiplier, $m = \frac{\sigma}{S}$. We chose the following values:

$$\begin{aligned} m &\in \{5 \times 10^{-4}, 10^{-3}, 5 \times 10^{-3}, 0.01, 0.05, 0.1, 0.5, 1.0\}, \qquad \text{for DP-SGD} \\ m &\in \{10^{-5}, 5 \times 10^{-5}, 10^{-4}, 5 \times 10^{-4}, 10^{-3}, 5 \times 10^{-3}, 0.01, 0.05, 0.1, 0.5, 1.0\}, \text{ for DP-Logits} \end{aligned}$$

C Further Results

In this section, we will present our results for the defense against the membership inference attacker for all the datasets, separately. Fig. 3 shows our results after application of DPSGD for both LRN and LRN-Free adversaries. The LRN adversary is a learning-based adversary adopted from [9] that utilized one shadow model and one attack model to perform the membership inference attack. Each bubble indicates a different noise level and the size of the bubbles are proportional to the noise. We can observe that by adding more noise the performance of the attacker drops (lower AUC values) but also the utility of the victim model decreases. The ideal attack would have an AUC=0.5 and accuracy close to the accuracy of the base classifier where no DP method is applied.

Fig. 4 shows the effect of applying DP-Logits. Again, different noise levels are shown with bubbles and the size of the bubbles is proportional to the amount of noise. Compared to DP-SGD we can see that the performance of the attacker drops faster for lower noise levels.

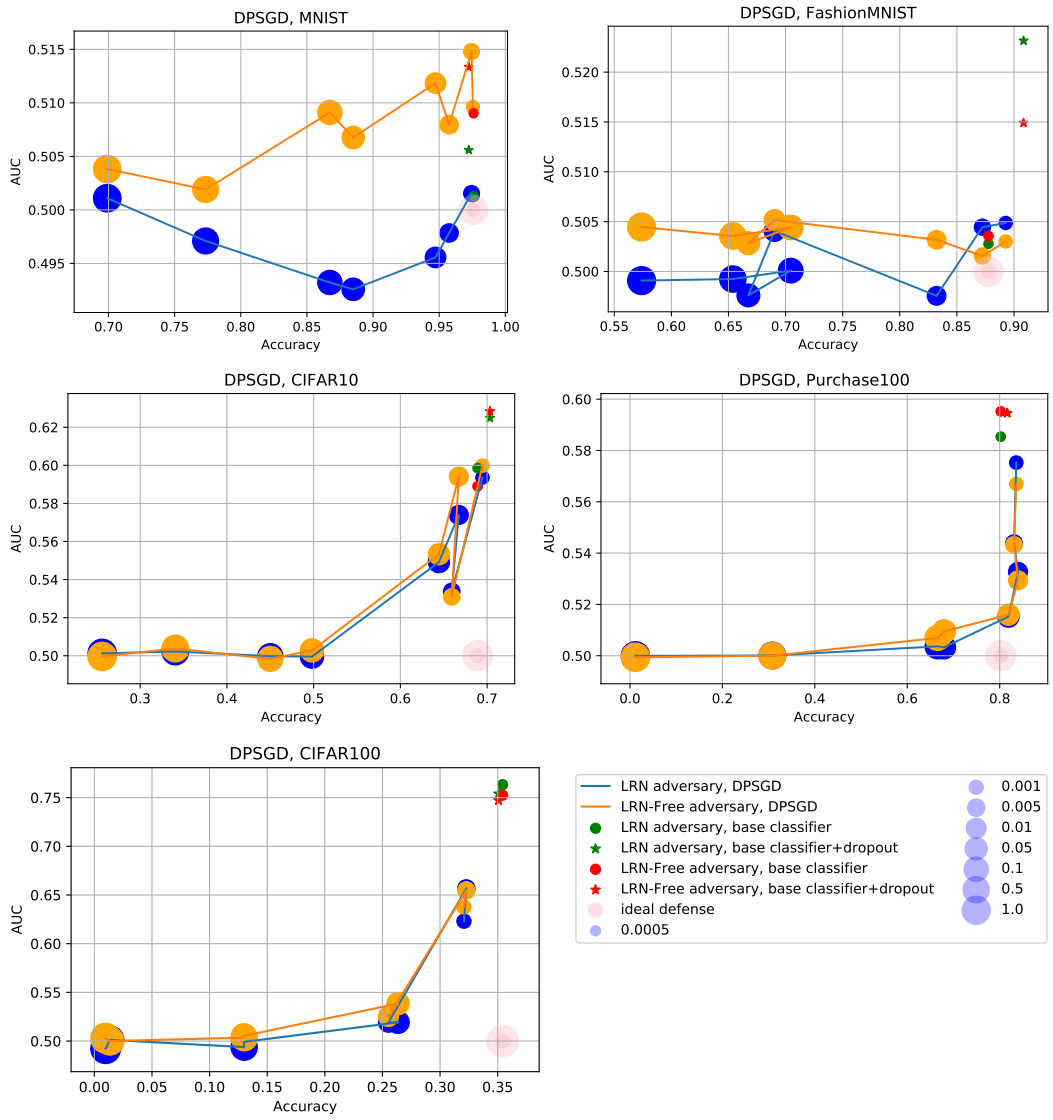


Figure 3: The effect of applying DPSGD during the training on all the datasets. We can observe that higher noise levels provide better protection against both adversaries but also reduce the accuracy of the victim model. Ideally, we are looking for a noise level with highest accuracy of the target model and lowest AUC value.

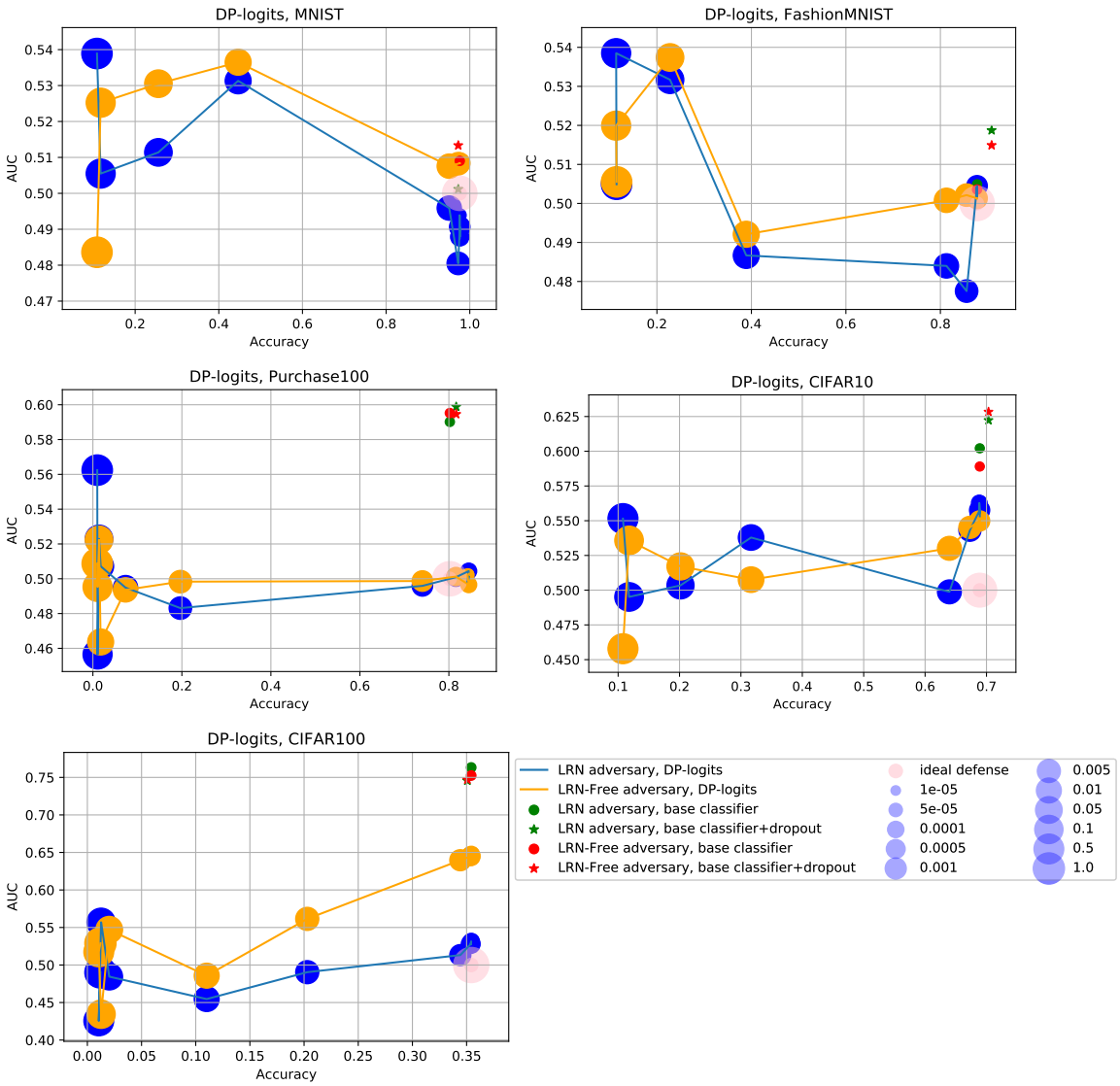


Figure 4: AUC versus accuracy of the target model for different datasets when the DP-Logits method is applied. The orange circles show the performance of the LRN-Free adversary and the blue circles show the performance of the LRN adversary. The size of the circles are proportional to the value of the noise multiplier. The pink circle shows the region of the ideal defense where the utility of the victim model is still high but attacker is completely unsuccessful.