SUPPLEMENTARY MATERIAL: LEVERAGING ADVERSARIAL EXAMPLES TO QUANTIFY MEMBERSHIP INFORMATION LEAKAGE

Appendix A describes our proposed ensemble membership inference attacker. In Appendix B we provide further experimental details and complementary results to those presented in the main paper, including results for several attack strategies that were initially considered, but under-performed. Finally, Appendix C provides additional results for the ensemble membership attacker.

A. Ensemble Attacker

This attacker requires not only white-box access to the model, as it needs to compute gradients with respect to input and to model parameters, but it also requires a training set of its own (similarly to [41, 45, 50]). Essentially, what the attacker learns is how to map different observations to a membership label.

The attack model is a DNN with 5 fully connected layers with output sizes $40, 40, 20, 10$ and 1, respectively. The input to the network is a vector of length 6, containing the softmax response, modified entropy, loss value, gradient norm w.r.t. parameters, gradient norm w.r.t. input, and adversarial distance. These quantities are re-scaled to $[0, 1]$, which significantly improves the performance of the model. The rescaling is done according to the maximum and minimum values from the training set. The model is trained with Adam optimizer [27] for up to 300 epochs. The performance of the ensemble attacker is evaluated and compared to the performance of other strategies in Tab. 5. Additionally, we vary the size of the attacker's training set and observe how this affects its performance. This results are presented in Tab. 6.

B. Further Experimental Details and Results

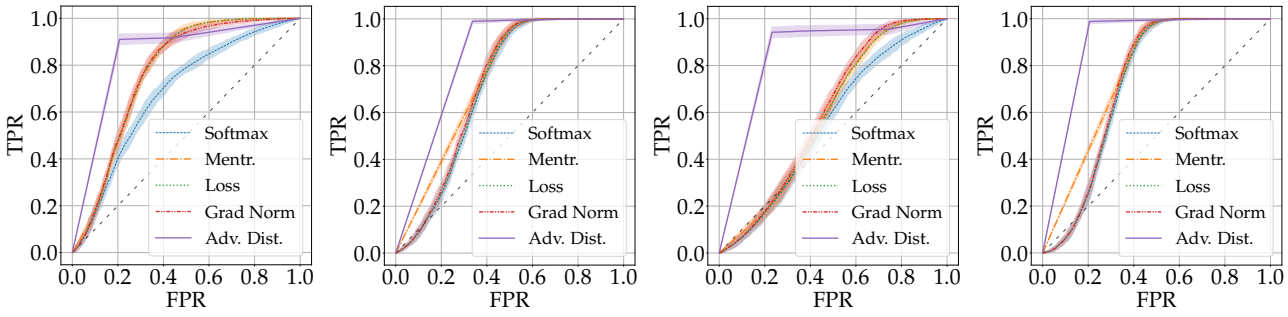
B.1. Experimental Details

Most of the experiments were run on a cluster with multiple nodes, each with NVIDIA Quadro RTX 6000 GPUs and an AMD EPYC 7302 16-Core processors.

When computing adversarial examples, we rescale the images so that their dynamic range lies within $[0, 1]$. This is necessary in order for the adversarial attacks to compute distance and perform clipping properly. However, since the pre-trained models were trained on the natural images (previous to rescaling), we include an additional layer at the input of each target model that reverts the scaling, preserving the performance of the target model.

The accuracy presented in Tab. 1 and Tab. 5 is computed by choosing a threshold along the ROC curve for each strategy. The threshold is chosen in order to maximize the accuracy. A similar process is done in the case of Tab. 2, where 80% of the data is used to determine the threshold that maximizes the accuracy, and then the accuray is reported for the other 20% of the data.

B.2. Additional Results



(a) AlexNet (b) DenseNet (c) ResNet (d) ResNext

Figure 4. ROC curves averaged over 20 iterations on AlexNet (4a), DenseNet (4b), ResNet (4c), ResNext (4d). The confidence interval correspond to 10 time the standard deviation.

Figure 4 is a copy of Fig. 3, except it includes the confidence intervals for the TPR. The curves in these figures are computed using the following process: A grid of FPR values is fixed. For each run of the experiment, we compute a ROC

Attack	AlexNet		ResNet		ResNext		DenseNet	
Strategy	AUROC	Accuracy	AUROC	Accuracy	AUROC	Accuracy	AUROC	Accuracy
Softmax	68.00 ± 0.16	65.34 ± 0.14	55.45 ± 0.15	57.40 ± 0.13	72.37 ± 0.07	74.84 ± 0.11	70.52 ± 0.09	72.11 ± 0.10
Mentr. [51]	77.11 ± 0.10	74.16 ± 0.11	59.10 ± 0.13	61.39 ± 0.11	76.87 ± 0.08	75.28 ± 0.11	74.21 ± 0.10	72.69 ± 0.10
Loss	76.69 ± 0.10	74.14 ± 0.13	58.66 ± 0.13	61.29 ± 0.11	72.57 ± 0.07	75.17 ± 0.11	70.85 ± 0.09	72.61 ± 0.10
Grad w Norm	76.58 ± 0.10	74.19 ± 0.12	59.93 ± 0.13	62.56 ± 0.09	73.06 ± 0.07	75.74 ± 0.11	71.30 ± 0.09	73.81 ± 0.09
Grad x Norm	75.20 ± 0.14	73.12 ± 0.12	59.64 ± 0.14	62.17 ± 0.11	72.92 ± 0.07	75.62 ± 0.10	71.15 ± 0.08	73.00 ± 0.09
Adv. Dist. $\|\cdot\|_\infty$	84.35 ± 0.13	85.12 ± 0.18	84.53 ± 0.16	85.45 ± 0.11	89.24 ± 0.03	89.10 ± 0.05	82.76 ± 0.03	82.63 ± 0.05
Adv. Dist. $\|\cdot\|_2$	76.89 ± 0.16	74.03 ± 0.15	70.86 ± 0.19	67.71 ± 0.18	72.47 ± 0.19	67.68 ± 0.16	68.00 ± 0.21	62.65 ± 0.20
Adv. Dist. $\|\cdot\|_1$	73.66 ± 0.12	74.03 ± 0.10	64.94 ± 0.16	64.51 ± 0.13	70.98 ± 0.19	66.95 ± 0.15	59.39 ± 0.17	59.27 ± 0.04
ML Attacker*	90.84 ± 0.13	85.48 ± 0.67	89.31 ± 1.04	85.07 ± 0.47	92.30 ± 0.19	92.17 ± 0.15	87.86 ± 0.22	87.46 ± 0.20
Grad w* [45]	78.76 ± 0.30	74.32 ± 0.28	61.98 ± 0.38	62.72 ± 0.27	77.80 ± 0.30	73.47 ± 0.57	73.12 ± 1.42	72.59 ± 0.55
Grad x* [45]	77.20 ± 0.26	73.43 ± 0.26	68.48 ± 0.27	63.58 ± 0.22	77.54 ± 0.61	73.47 ± 0.57	75.81 ± 0.43	71.81 ± 0.40
Int. Outs* [45]	57.92 ± 0.50	56.36 ± 0.41	96.59 ± 0.29	91.57 ± 0.43	93.62 ± 0.39	86.38 ± 0.37	99.17 ± 0.10	97.68 ± 0.14
Logits* [45]	58.19 ± 0.57	56.35 ± 0.52	88.96 ± 0.44	81.38 ± 0.57	84.89 ± 0.37	77.24 ± 0.33	67.64 ± 0.57	63.19 ± 0.46
WB* [41]	80.33 ± 1.21	74.03 ± 0.71	87.51 ± 0.41	79.73 ± 0.30	84.52 ± 1.95	76.46 ± 1.82	79.38 ± 1.16	71.92 ± 0.97

Table 5. Comparison of different MIA Techniques. The Accuracy(%) and AUROC score (%) on a balanced evaluation set are reported. $10k$ are uniformly selected from the training set (members) and the whole $10k$ samples from the testing set are selected (non-members). All the data selected is used for evaluation. Techniques with a (*) require training. In this case, only 60% of the data is used for evaluation and rest is used for training.

curve and use it to interpolate the TPR values that correspond to the fixed FPR values. The TPR values are averaged over different runs of the experiment.

The confidence intervals presented in Fig. 4 correspond to 10 times the standard deviation. We chose to multiply the standard deviation in order to make the confidence intervals visible in the figure. Indeed, in our experimental setting as the train and test sets have large sizes, the ROC curves of MIA strategies remain close to the average ROC curves over several shots of train and test sets.

Table 5 contains the results for additional attack strategies that were not included in the main body of the paper.

Among the strategies that can be implemented as a binary decision test, we include the Grad x Norm strategy, and the adversarial distance strategy with other different norms. In the Grad x Norm strategy, the gradient of the loss function of the target model with respect to the input image is computed. Then, the ℓ_2 norm of the gradient is computed and used for the binary decision test. The Grad w Norm is equivalent to the Grad Norm strategy presented in the main body of the paper. In the main body of the paper, we present the results for the adversarial distance strategy that uses the ℓ_∞ norm to measure the distance between samples. Here we include the results for the ℓ_1 and ℓ_2 norms.

Table 5 also shows the results for two additional models that require training. Namely, the ensemble attacker explained in Appendix A and referred to as ML attacker in the table, and the Logits attacker from [45]. The Logits attacker is similar to the intermediate outputs attacker explained in the main body of the paper, but only utilizes the outputs of the last layer.

C. Additional Results for the ML Attacker

We also study how the amount of side information influences the performance of the attacker. Table 6 reports the AUROC score of the ML Attacker against the pre-trained target models for different amounts of side information. The side information is always composed by 50% *in-training* samples and 50% *out-of-training* samples, and the total is indicated in the table. Remark that the performance of the ML Attacker might improve with the increase in the size of its training set; however, in some cases a small training set (1000 samples) is enough to obtain an effective attack model.

Target	Training set size			
Model	1000	2000	4000	8000
AlexNet	86.68 ± 1.95	88.25 ± 1.63	89.87 ± 0.73	90.84 ± 0.52
ResNet	87.15 ± 1.05	88.34 ± 0.68	89.26 ± 0.55	89.31 ± 1.04
ResNext	92.24 ± 0.22	92.30 ± 0.28	92.31 ± 0.19	92.30 ± 0.19
DenseNet	86.20 ± 5.51	87.42 ± 1.45	87.71 ± 0.56	87.58 ± 0.22

Table 6. Influence of training set size on performance for the ML Attacker. The AUROC (%) for a balanced evaluation set is reported. Half of the training samples for the attacker are uniformly selected from the original training set and the other half from the test set. Then, $6k$ samples from the training set and $6k$ samples from the test set are uniformly selected for evaluation.