Property Inference as a Regression Problem: Attacks and Defense

Joshua Stock[1][a], Lucas Lange[2][b], Erhard Rahm[2][c] and Hannes Federrath[1]

[1] *Security in Distributed Systems, Universität Hamburg, Germany*
firstname.lastname@uni-hamburg.de

[2] *Database Group, Universität Leizpig, Germany*
lastname@informatik.uni-leipzig.de

Keywords: Machine Learning, Privacy Attacks, Property Inference, Defense Mechanisms, Adversarial Training

Abstract: In contrast to privacy attacks focussing on individuals in a training dataset (e.g., membership inference), Property Inference Attacks (PIAs) are aimed at extracting population-level properties from trained Machine Learning (ML) models. These sensitive properties are often based on ratios, such as the ratio of male to female records in a dataset. If a company has trained an ML model on customer data, a PIA could for example reveal the demographics of their customer base to a competitor, compromising a potential trade secret. For ratio-based properties, inferring over a continuous range using regression is more natural than classification. We therefore extend previous white-box and black-box attacks by modelling property inference as a regression problem. For the black-box attack we further reduce prior assumptions by using an arbitrary attack dataset, independent from a target model's training data. We conduct experiments on three datasets for both white-box and black-box scenarios, indicating promising adversary performances in each scenario with a test R² between 0.6 and 0.86. We then present a new defense mechanism based on adversarial training that successfully inhibits our black-box attacks. This mechanism proves to be effective in reducing the adversary's R² from 0.63 to 0.07 and induces practically no utility loss, with the accuracy of target models dropping by no more than 0.2 percentage points.

1 INTRODUCTION

Machine Learning (ML) technologies are as present as never before and their advancement is significantly enhancing capabilities across multiple domains. However, this progress also introduces new challenges, particularly in the realm of data privacy. In healthcare and other user-centric areas, training large ML models requires ever increasing amounts of personal data to provide the desired outcomes. Dangers to privacy mostly stem from ML models being prone to leaking information about their training data under adverserial attacks (Al-Rubaie and Chang, 2019). A famous example among these are membership inference attacks (Shokri et al., 2017), deciding membership of individuals in training data, given a trained model. More recently Property Inference Attacks (PIAs) (Ateniese et al., 2015) have emerged as a critical threat, where adversaries aim to infer sensitive properties from the training data of target models without direct access to the underlying data. Examples for such critical properties include the gender ratio (male to female) in a dataset or the status of security patches in log files used by an intrusion detection system (Ganju et al., 2018). These adverserial threats lead to an increasing body of research in Privacy-Preserving ML (PPML) that revolves around attacking and defending ML models in various scenarios (Xu et al., 2021).

The threat of PIAs is especially severe in in white-box scenarios, where attackers have complete knowledge of the target model's architecture and parameters (Ateniese et al., 2015; Ganju et al., 2018; Stock et al., 2023). However unlike their white-box counterparts, black-box attacks do not require detailed knowledge of the model, reflecting more realistic adversarial scenarios against services that only query models for their outputs (Zhang et al., 2021). We introduce a novel method of training a black-box adversary for property inference, where we extend the conventional black-box PIAs to use an arbitrary at-

[a] https://orcid.org/0000-0003-3940-2229
[b] https://orcid.org/0000-0002-6745-0845
[c] https://orcid.org/0000-0002-2665-1114

tack dataset as input to the target model, which can be independent from the model's training data. By formulating PIAs as a regression problem, the adversary can freely extract the most likely distrubtion of a property in the original training data (Suri and Evans, 2022). We show the feasibility of our black-box PIA in attacking Random Forest models and Deep Neural Networks (DNNs). To put our black-box results into context, we compare them to an adapted white-box attack by (Ganju et al., 2018) that we run for the same scenarios. In all scenarios, we are able to obtain adversaries with a coefficient of determination (R^2) of at least 0.6, ranging up to 0.72 with black-box access and up to 0.86 with white-box access.

As a defense mechanism to our black-box PIA attack, we test the possiblity of directly including an adverserial loss in model training, which is inspired by (Grari et al., 2020) who showed the effectiveness for enhancing a model's fairness. The idea is to check the model's proneness to PIAs after each training round and to then accordingly influence the model's next updates to steer the model into hiding the original property distibution from its outputs. The model is instead trained to blind the adversary with a predefined target property value. We evaluate the resulting ability on mitigating the risks associated with black-box PIAs but also monitor the negative impact on model utility from impairing model training. Our results demonstrate that our mechanism effectively reduces black-box PIA success by an order of magnitude to an R^2 of 0.07, while only incurring a negligible utility loss of less than 0.2% accuracy.

We summarize our contributions as follows:

- We expand current PIAs for the black-box setting by enabling our meta-classifier to use an arbitrary attack dataset, which is indepedent from the target's original data distribution.
- For this meta-classifier, we frame our property inference as a regression problem that is able to extract a more realistic range of sensitive property distributions compared to the predefined distributions of a classification task.
- As a new defense mechanism we propose an adverserial training scheme that hides the actual property distribution from attacks by guiding the model during training to produce more balanced outputs, while preserving model utility. In contrast to defense schemes in related work, our method generalizes well, i.e., defends against a whole class of PIAs instead of defeding against a single adversary instance.

For organizing this work, Section 2 provides an introduction to important related work on PIAs and possible defense mechanisms. Section 3 describes our methodology for conducting black-box PIAs and framing these attacks as a regression problem. Section 4 then outlines the proposed countermeasures for mitigating PIAs. In Section 5, we present our experimental setup, including the datasets used, the models evaluated, and the metrics for assessing the success of attack and defense. Section 6 discusses the results of our experiments, demonstrating the effectiveness and limitations of our approaches, and suggests future research directions. Finally, Section 7 concludes with a summary of our findings.

1.1 Performance metrics

Since we model the adversarial task as a regression problem, we measure the adversary's success with the *coefficient of determination (R^2)*. A dummy regressor always outputting the expected value of the trained labels would yield an R^2 of 0, while an ideal regressor's predictions would amount to an R^2 of 1.

In our evaluation, we use *boxplot graphs*, since they capture multiple characteristics of a distribution: the average is plotted as a diamond, while a line within the box is the mean of the distribution. The upper and lower line of the boxes capture the first and third quantile, while the whiskers below and above the box imply the range of values outside the two main quantiles. Outliers are plotted as points above or below the whiskers.

2 RELATED WORK

The task of property inference, also known as distribution inference, is first introduced by (Ateniese et al., 2015) who describe attack patterns on traditional machine learning models. The current state-of-the-art white-box PIA by (Ganju et al., 2018) is based on permutation-invariant networks and adapts earlier PIAs with a focus on fully connected neural networks. Their solution has since found successful adoption to other ML models and is the main focus for evaluating white-box PIAs (Hartmann et al., 2023; Suri and Evans, 2022). In the same vein, we find the first approach for black-box attacks introduced by (Zhang et al., 2021) to form the basis for the current implementations of black-box PIAs in these works. Their attack method uses a set of shadow models each trained on data exhibiting different distributions of the attacked property. They then train a meta-classifier that learns to predict the property of interest based on the shadow models' outputs on a query dataset. During execution, the attacker queries the target model with the attack query set, feeds the outputs to the meta-classifier, and obtains a prediction of the sensitive property in the data. The standard attack setting for PIAs is a binary classification over two predefined distribution values to infer to which of them the dataset property conforms. For a more realistic case of continuous distribution values like for the gender ratio, other works extend the inference task from binary to a larger set of possible distributions (Zhang et al., 2021; Zhou et al., 2022; Suri and Evans, 2022).

For defending against PIAs, (Hartmann et al., 2023) try to evaluate the fundamental reasons for why models leak sensititve information about their distribution. They come to believe that three factors play an intervening role in enabling such exploits. As solutions they conversly suggest to reduce the model's memorization about the expected label given the features of adversary interest, optimizing model architecture against wrong inductive bias, and increasing the amount of training data. While these mitigations may help, they each come at a cost. Reducing memorization impacts model performance, optimizing model architecture requires prior assumptions, and collecting more training data is not always feasible.

Other methods for protecting against PIAs also have shown only limited effectiveness in reliably preventing their success. In contrast to membership inference attacks, differential privacy techniques are not effective at mitigating inference risks related to data distributions as tested by (Ateniese et al., 2015). This is because differential privacy focuses on obfuscating the contributions of individual records, whereas in the PIA setting, the adversary is able to uncover statistical properties of the underlying data distribution on a population level. Another approach by (Zhang et al., 2021) tries to remove sensitive attributes from datasets, which has also proven ineffective, as the correlations between different attributes still prevails. (Ganju et al., 2018) propose a defense using scaling transformations applied in models with ReLU activations that hide their learned distributions in the internal weights. However, this technique does not offer any protection against black-box attacks. (Zhou et al., 2022) suggest alterations to the training data with respect to the target property before training the model. This can be done in two ways: either by removing records or by adding new records. By this, the ratio of a property should reach a predetermined fixed value, such as 0.5, regardless of the original ratio. However, this strategy would either strive for acquiring new data or removing records from the dataset to balance the property ratio, leading to expensive data acquisition or a potential reduction in training data. In (Stock et al., 2023), they test property unlearning as a defense mechanism against white-box attacks. It uses an adversarial classifier to identify the model parameters that leak information about a property. They then use backpropagation to modify them into unlearning the property. Experiments show this can be effective against a specific white-box adversary, but has limitations in generalizing and can therefore not protecting against black-box or different white-box PIAs targeting a certain property. In summary, the currently available defences against property inference still have significant limitations and do not provide reliable protection across various attack scenarios.

To address the limitiations stemming from related work, we first take the black-box PIA approach by (Zhang et al., 2021) and shift the attack set to be independent from the original data distribution to further reduce the burden of pre-requisites on black-box PIAs. We further adopt the suggestion by (Suri and Evans, 2022) and take property inference as a logistic regression problem to directly infer specific property ratios through our meta-classifier, where others only consider a definite set of distributions and formulate the adversarial task as a classification problem. For defending against black-box PIAs, we adapt an adversarial learning strategy introduced by (Grari et al., 2020), where they directly include an adverserial loss term during model training to influence its memorization regarding specific properties. In their work they utilize this training framework to achieve fairness in decision tree models. Transitioning this process to PIA defense, we use a PIA adversary during training to punish a model, when it memorizes a distribution too much, leading to an increased adversary loss term. Using this process, we can dictate a model to learn a specific distribution output, and to hide its true distribution, regarding a sensitive property. Our goal in defending is comparable to the proposition by (Zhou et al., 2022) but is applicable without actively removing training samples and we instead just reduce their influence to steer the property exposure.

3 REGRESSION PROPERTY INFERENCE ATTACKS

Given a trained target model, the overall aim of a PIA is to recover sensitive information about the model's training data. In contrast to other attacks, PIAs are not directed at properties of an individual data sample but rather at *global* properties of the training dataset. Examples for such critical properties include the gender ratio (male to female) in a dataset or the status of security patches in log files used by an intrusion detection system (Ganju et al., 2018). While some properties are binary, this work focuses on the more difficult problem of extracting continuous properties. Hence, we introduce PIAs as a regression problem, as proposed in (Suri and Evans, 2022).

3.1 Black-box attacker model

In this work, we focus on a black-box attacker model, in which an attacker does not have access to the target model itself, but can choose input values and observe the target model's output. This is a realistic scenario for many applications in which machine learning models are deployed as a service. In real-world applications, the internal model weights are often hidden from clients and only API-access is granted, meaning that requests are forwarded to the model internally and clients only receive its output. This helps the model owner to stay in control, and attacks such as model stealing (Tramèr et al., 2016) can be prevented more easily.

The attacker has some information about the target model's training data or can access parts of it. This is an assumption also made for previous defense strategies (Nasr et al., 2018; Song and Mittal, 2021; Tang et al., 2021; Stock et al., 2023). Otherwise, information about the training data can also be reconstructed with separate attacks (Shokri et al., 2017), which is just as effective (Liu et al., 2022).

3.2 Attack Description

For our black-box PIA execution, we adapt the techniques from (Zhang et al., 2021) for our scenario. The attack can be split in four steps, as described below and depicted in Figure 1.

Step 1: Training shadow models

Since the ultimate goal of an adversary $\mathcal{A}$ is to predict a property for a target model, $\mathcal{A}$ trains on the output of models with known property values – the shadow models. Hence, the first step in a PIA is to train multiple shadow models for different property values, i.e., shadow models are trained on training datasets with the respective property values. Since the adversary $\mathcal{A}$ is modeled as a regressor, we first create auxiliary training datasets with the property values $x \in [0.1, 0.2, \ldots, 0.8, 0.9]$, i.e., each auxiliary training dataset DS^x_{aux} features ratio x regarding a predefined property. As an example, the property might be defined over the ratio of men to women in a dataset. To create an auxiliary dataset DS^x_{aux} in our experiments, we use the original training dataset and delete records until property value x is reached. This process is repeated for all $x \in [0.1, 0.2, \ldots, 0.8, 0.9]$, such that the resulting 9 auxiliary training datasets DS^x_{aux} can be used to train shadow models. On each DS^x_{aux}, k shadow models are trained. For the datasets we have used in our experiments, $k = 200$ has proven useful. While a higher number for k might increase the utility of the advesary, it comes with higher computational costs.

By basing the auxiliary datasets on the original training datasets, we create worst-case adversaries, i.e., the strongest possible adversaries: Since the distribution of the DS^x_{aux} is as close as possible to the original training data distributions, the adversary learns the behaviour of very similar models to the attacked model. This is especially helpful when model owners defend their models during training. Since original training data must be available to the model owner during the training process, this prerequisite is easily fulfilled.

Step 2: Generating output from shadow models

The focus of this work is on a black-box attacker model, meaning that the adversary $\mathcal{A}$ does not have access to a target model itself but can only survey its output on a chosen input. Therefore, after training the 9*200 shadow models, they are each queried with the same input. This input can be arbitrary, as long as it yields a meaningful output when used as input to the shadow models. Since this input is an essential part of the black-box attack, we call it attack dataset DS_{att}. The output of the shadow models, labeled with the according property values $[0.1, 0.2, \ldots, 0.8, 0.9]$, is then stored as the training dataset for the adversary $\mathcal{A}$. The same process of training and generating output on shadow models is repeated on equally sized and non-overlapping auxiliary *test* datasets. For creating a test dataset for the adversary, we use 50 shadow models per property value, i.e., in total there are 9*200 shadow models to generate the meta training set and 9*50 shadow models to generate the meta test set.

Step 3: Training adversary on shadow model output

The adversary $\mathcal{A}$, or *meta-classifier*, is modeled as a simple deep neural network (DNN). The training dataset of the adversary (*meta training set*) consists of the outputs of the trained shadow models on the chosen input data, paired with the property values of their respective training datasets. The hyperparameters of the adversary such as neurons per layer and batch size are optimized via the framework keras-tuner[1]. The optimizer Adam is used to fit the model and early

[1] https://keras.io/keras_tuner/

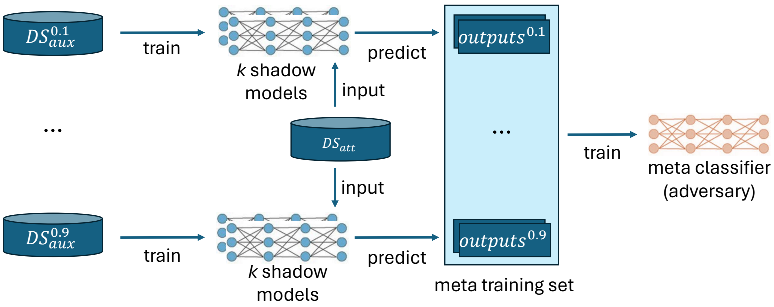

Figure 1: Black-box PIA in three steps: Train shadow models, generate output and train adversary.

stopping is applied to avoid overfitting. After approximately 65 epochs, the meta-classifier $\mathcal{A}$ reaches its peak test R² value and the training stops.

Step 4: Attacking target model

After training the adversary $\mathcal{A}$, a target model m can be attacked by querying it with the attack dataset DS_{att}. The output of m then serves as the input for the adversary to infer the sensitive property, i.e., $\mathcal{A}(m(DS_{\text{att}})) = x$, where x is the value of the sensitive property inferred by the adversary $\mathcal{A}$.

3.3 White-Box Benchmark PIA

In order to benchmark our novel black-box regression adversaries described above, we train white-box advesaries on the same datasets for comparison. In principle, we follow the work of (Ganju et al., 2018), which is also based on shadow models. Hence, step 1 in Section 3.2 is identical in the white-box setting. Step 2 (generating output from shadow models) is not necessary, since the adversary is trained on the shadow model weights and biases itself – constituting the final step of training a white-box adversary. The white-box attack is carried out by directly using the weights and biases of a target model as input to the trained adversary.

4 DEFENDING BLACK-BOX PIA

To defend against black-box PIAs, we propose a form of adversarial training. We borrow the technique of (Grari et al., 2020), originally intended to make models *fair*, in the sense of making decisions independent from pre-specified sensitive characteristics in the data. Grari et al. design an adversary with the task of deducing the sensitive data property from the model output for Gradient Boosted Trees (GBTs). Using this adversary during training, they create additional gradients for the trained model – to minimize the success of the adversary, i.e., to minimize the influence of the sensitive property on the model's output. A parameter λ is introduced which controls the tradeoff between the two competing training goals model performance and property suppression.

We have extended the strategy of (Grari et al., 2020) for GBTs to defend DNNs against PIAs. For defending a target DNN during training, we can simply modify its loss function. To be precise, we define two terms within the loss function: A first term punishing the model if its prediction deviates from the labels of the training data, and a second term punishing the model if the PIA adversary infers another value than the predefined target distribution from the model's output. As in the work of (Grari et al., 2020), the two terms are weighted with $(\lambda - 1)$ and λ, allowing to find an optimal tradeoff between model optimization and PIA defense.

When computing the loss ℓ as a mean squared error (mse), this yields

$$\ell = (1-\lambda) * mse(y_{true}, y_{pred}) + \lambda * mse(adv_{tar}, adv_{pred})$$

for training data labels y_{true} and model predictions y_{pred}, the adversary's target distribution adv_{tar} and the adversary's current prediction adv_{pred}) based on the model's output.

5 EXPERIMENTS

We perform our experiments on the three datasets Adult, UTKFace and CIFAR-10. We have performed our experiments on linux machines using Python 3.10 and the latest versions of TensorFlow and keras. The defense experiments have been performed on 10 target models per property value and dataset. The target models are trained on a different portion of the datasets than the shadow models (i.e., the auxiliary test datasets as described in Section 3.2, Step 2) to avoid side effects.

5.1 Datasets

Adult is a dataset consisting of 32,561 records from the US census (Kohavi et al., 1996). The 14 attributes include information about the individual gender, work hours per week and education level. The machine learning task is to predict whether an individual earns more than 50k dollars per year. For this tabular dataset, we use random forest classifiers, reaching an accuracy of 85%.

As the sensitive attribute, we choose the attribute *gender* in the adult dataset, i.e., the adversary's goal is to predict the ratio of male:female records in the training dataset. For the *attack dataset* which the adversary uses to generate output from target models, we generate a synthetic dataset with 10,000 records based on Adult using a conditional generative adversarial network (CTGAN) (Xu et al., 2019).

UTKFace consists of 20,000 color images with annotations of age, gender and ethnicity (Zhang et al., 2017). We choose the ML task of inferring the correct gender from an image. Deep neural networks with convolutional layers are used for recognizing the images, reaching a test accuracy of 90%.

We define our sensitive property over the *age* attribute, i.e., the property is defined by the distribution of old to young instances in the data. For simplicity, we consider all *age* labels above 59 as old. Our attack dataset is based on the *Labeled faces in the wild* dataset (Huang et al., 2008), containing 13,233 images of 5,749 different people.

CIFAR-10 contains 60,000 color images for image recognition tasks with 32x32 pixels each (Alex, 2009). The dataset is grouped into 10 classes, such as *airplane*, *horse* and *truck*. For the target and shadow models, we use deep neural networks with the same architecture as for UTKFace above, reaching accuracies of 70% (random guessing would amount to 10% accuracy).

The sensitive attribute we define in this dataset is the amount of animals in the training dataset, i.e., the adversary's task is to predict the ratio of animal:non-animal images in the training dataset of the target model. The attack dataset which the adversary uses to generate output from the target model is based on the CIFAR-100 dataset (Alex, 2009). We randomly select 5,040 samples of 21 CIFAR-100 classes, which share similarities with CIFAR-10 classes, e.g., images of the CIFAR-100 class *lion* have similar features as the CIFAR-10 class *cat*.

Access	white-box	black-box	
Defense	none	none	$\lambda = 0.15$
Adult	–	0.72	–
UTKFace	0.86	0.63	0.070
CIFAR-10	0.60	0.64	0.069

Table 1: Adversary performance measured as R² on test data, comparison black-box and white-box. Both the white-box attack and the black-box defense are not applicable to the tree-based models for the Adult dataset.

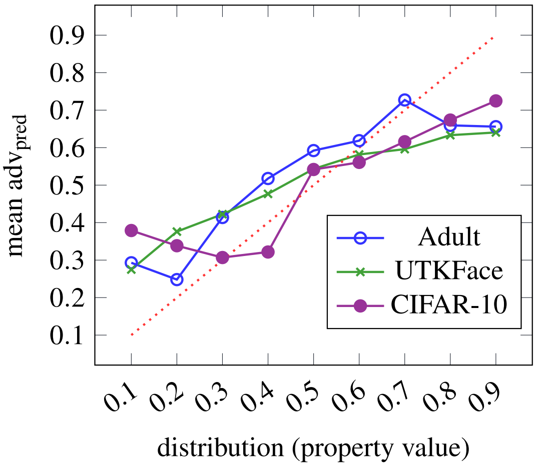

Figure 2: Performance of black-box adversaries, mean adversary output on test data for different distributions (resp. property values).

5.2 Regression PIA results

The results of our attack experiments are summarized in the first two columns of Table 1. Following the regression approach explained in Section 3, we have trained PIA adversaries both in white-box and black-box scenarios[2]. The black-box adversaries reach R² values from 0.63 (UTKFace) to 0.72 (Adult). Their performance is visualized in Figure 2: An ideal adversary would infer all property values correctly, as shown by the dotted red line. It is apparent that all three adversaries perform best in the mid-ranges of property values (0.2–0.7), where their predictions are closest to the ideal. For CIFAR-10, the largest deviation is at property value 0.1, where the mean prediction amounts to 0.38. The smallest deviations are observed for property value 0.6, where all three adversaries' mean predictions deviate less than 5% from the correct value.

Focusing on the first column of Table 1, we can see that the white-box adversary for UTKFace outperforms its black-box counterpart (R² of 0.86 com-

[2] The white-box approach was not applicable to the GBTs of the adult dataset, since the proposal in (Ganju et al., 2018) is designed for neural networks.

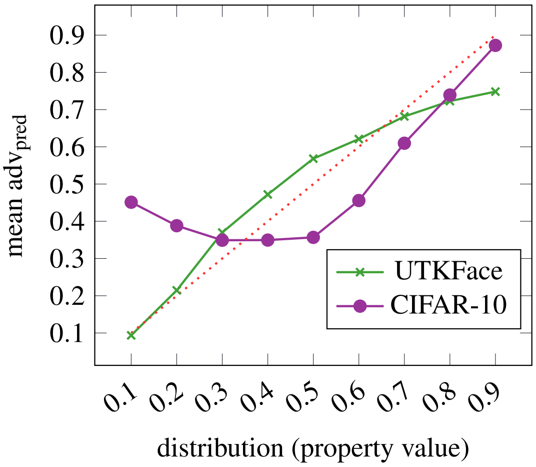

Figure 3: Performance of white-box adversaries, mean adversary output on test data for different distributions (resp. property values).

pared to 0.63), while the performance of the white-box adversary for CIFAR-10 is worse than the black-box version (R^2 of 0.60 opposed to 0.64). The great performance of the white-box UTKFace adversary is also reflected in Figure 3, where the plotted deviations from the ideal predictions (dotted red line) are small or even not existing in the case of property value 0.1. At the same time, the CIFAR-10 white-box adversary performs worst at property value 0.1, its mean prediction deviating even more from the truth than its black-box counterpart in Figure 2. Interestingly, the CIFAR-10 white-box adversary obtains its best predictions for property value 0.9, contrary to the UTKFace white-box adversary.

5.3 Regression black-box PIA defense results

We have implemented the defense strategy of Section 4 and conducted experiments for different values of λ. To recapitulate, the higher the value of λ, the more the trained model is defended against a PIA adversary, hence $\lambda = 0$ implies a regular training without any defense mechanisms. All experiments were run with the target property value 0.5.

To show the effect of our adversarial training, we plot the adversarial outputs after defending target models with $\lambda = 0.15$ in Figure 4: The plot is very different from the original adversary performance in Figure 2, with mean adversary outputs close to the target 0.5 for target models with all property values. This underlines the results in the third column of Table 1 with anadversary R^2 of 0.07 on defended models for both datasets UTKFace and CIFAR-10.

More detailed results for different values of λ are presented in Figure 5 for UTKFace and Figure 6 for the CIFAR-10 dataset. The green boxplots at the bottom of both figures represent the distance of adversary outputs to the target value 0.5 – as λ increases along the x-axis, this distance decreases. The black boxplots at the top of the figures show the accuracies of the defended models. For both datasets, we can observe practically no accuracy decrease: The mean accuracy even increases for the UTKFace models from 86.19% ($\lambda = 0$) to 86.60% ($\lambda = 0.15$), before it decreases slightly to 84.91 for $\lambda = 0.2$. For CIFAR-10, the accuracy decreases slightly from 67.46% ($\lambda = 0$) to 67.33% ($\lambda = 0.25$).

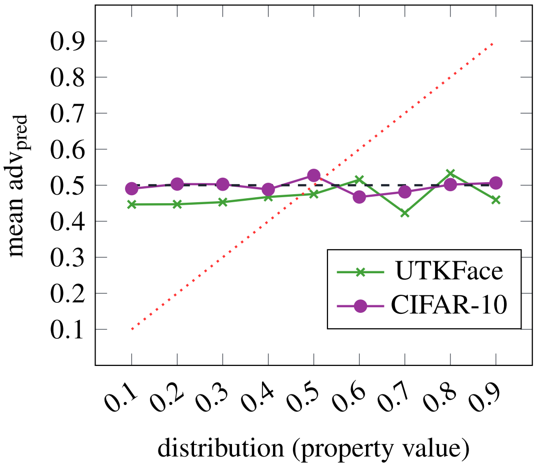

Figure 4: Black-box adversary outputs for target models defended with $\lambda = 0.15$ and target property value 0.5.

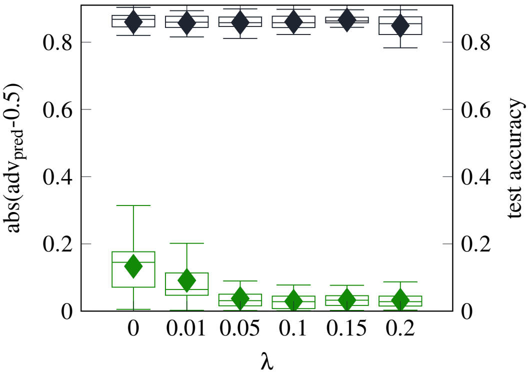

Figure 5: UTKFace: Accuracy of defended models for different values of lambda (black) and absolute divergence from target 0.5 of the adversary for defended models (green).

6 DISCUSSION

Regarding the attack success rates, the black-box and white-box adversaries are in the same range of 0.6 to

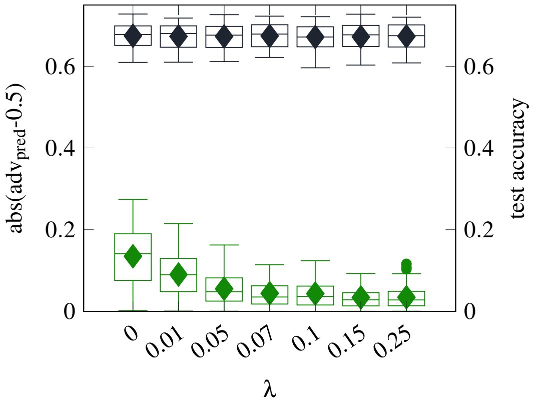

Figure 6: CIFAR-10: Accuracy of defended models for different values of lambda (black) and absolute divergence from target 0.5 of the adversary for defended models (green).

0.86 R^2. While the *age* property for the UTKFace dataset is easily extracted in the black-box scenario (highest R^2: 0.86), the *animal* property of CIFAR-10 seems to be harder to extract for an adversary, especially for property values below 0.3 (both in the black-box and white-box scenarios). The reasons behind this are guesswork; one hypothesis could be that as long as the share of animal images during CIFAR-10 training is below a certain threshold, the neural network does not account for typical animal features to an extend which is inferrable for an adversary.

In general, the attack success rates are hard to compare to related work, since PIAs are only considered as a classification problem, not covering the more natural continuous value range of property distributions. To the best of our knowledge, (Zhang et al., 2021) is the only work with a similar PIA experiment. In reference to their "fine-grained" attack using 5 classes for the property *gender* in the Adult dataset, we calculate approximate accuracy values for our attackers by classifying our adversary outputs into 4 classes across the range 0–1. The value from (Zhang et al., 2021) is the average value of the rightmost column in Table 7, i.e., the average accuracy across the 5 classes. The results (see Table 2) exhibit similar accuracy rates across all adversaries (39.8%–67.6%), although Zhang et al.'s work seems to outperform our black-box attacks. The two white-box attacks show significantly higher accuracy rates than the black-box attacks. However, we need to stress that the accuracy values are extracted from our regression outputs and that our adversaries have not been optimized to achieve high accuracies in a classification task.

In our defense experiments, we have shown that the adversarial approach, actively suppressing property information in target model outputs, works well and does not negatively affect the performance of defended target models. Across both datasets, $\lambda = 0.15$ has proven to create a reasonable tradeoff during training, minimizing the adversary's R^2 to 0.07 while harming the target models' performance by less than 0.2 percentage points on average. As Figure 5 exhibits, using a higher λ than 0.15 is not necessary, since the difference in adversary performance is negligible (R^2 of 0.067 for $\lambda = 0.25$ instead of R^2 of 0.069 for $\lambda = 0.15$) and two outliers imply possibly unwanted behavior for higher values of λ. Another notable observation from Figure 5 and Figure 5 is that not only the accuracy values are stable, but also their deviations do not increase for bigger values of λ.

	black-box		white-box
	related work	ours	ours
Adult	50.6%	39.8%	–
UTKFace	–	44.9%	67.6%
CIFAR-10	–	49.2%	62.4%

Table 2: Approximated accuracies for our attacks to establish comparability to related work. The value for related work is extracted from (Zhang et al., 2021), Table 7.

For demonstration purposes, we have used 0.5 as the target property value across all defense experiments. In practice, this value could be either randomly chosen from the range 0–1 for each target model individually, or set to some other constant value. Depending on the use case, one option might make more sense than the other.

In (Stock et al., 2023), the authors have demonstrated how it does not suffice to harden a target model against a single white-box adversary to defend a whole class of PIA adversaries. For black-box adversaries, this is different, since the information available to the adversary is a lot more sparse than in white-box scenarios (model output vs. all trained weights and biases). Therefore, adversaries cannot circumvent the adversarial defense by focusing on another part of available information, as has been shown for the white-box case in Section 6 of (Stock et al., 2023). We were able to confirm this experimentally by validating that defending a target model against one black-box adversary limits the capabilities of another adversary with the same task at the same rate. This shows that the defense mechanism in this work generalizes well, in contrast to the white-box defense presented in (Stock et al., 2023).

6.1 Future Work

Through our experiments, we have demonstrated the feasibility of regression PIAs. Although this will not prevent attackers from executing them, the process of implementing the attack takes a lot of effort, entailing the identification of a target model's training data distribution, creating shadow models, fine-tuning the architecture of the adversary model, etc. Follow-up work could investigate whether this effort could be limited, while maintaining the success rates shown in this work. Inspiration could be taken from (Li and Zhang, 2021), where a "boundary attack" for membership inference is presented, which bypasses the creation and usage of shadow models altogether.

Also, our defense mechanism could be transferred to a hybrid scenario. Instead of using a static adversary during the adversarial training of a target model, the adversary is retrained on the modified output of the target model in every epoch of the target model training, as in (Grari et al., 2020). Such a hybrid adversarial training could potentially further reduce the leaked property information of a target model, although further investigation is necessary.

Last but not least, the adversarial training for fairness in (Grari et al., 2020) could have side effects on the success rate of PIAs. Since Grari et al. train their models to yield outputs independent from a sensitive property p, it would be interesting to investigate whether their approach could also defend a target model against a PIA focussing on property p.

7 CONCLUSION

In this work, we have expanded upon existing black-box PIAs by using an arbitrary attack dataset, which can be based on other datasets than the training dataset. As the natural fit for many ratio-based properties, we have modeled the PIAs in this work as regression problems. We have explored a defense mechanism based on adversarial training which hardens a target model against black-box PIAs during its training process. We have evaluated our approach on three datasets, comparing the attack against white-box benchmarks and related work. In our experiments, we have shown our defense scheme to be both effective (by decreasing the adversary's performance from an R^2 of 0.63–0.64 to 0.07) and practical, decreasing the mean accuracy of target models by less than 0.2 percentage points.

ACKNOWLEDGEMENTS

The authors would like to thank Jona Zantz for his helpful comments and insights.

REFERENCES

Al-Rubaie, M. and Chang, J. M. (2019). Privacy-preserving machine learning: Threats and solutions. *IEEE Security & Privacy*, 17(2):49–58.

Alex, K. (2009). Learning multiple layers of features from tiny images. *https://www. cs. toronto. edu/kriz/learning-features-2009-TR. pdf.*

Ateniese, G., Mancini, L. V., Spognardi, A., Villani, A., Vitali, D., and Felici, G. (2015). Hacking smart machines with smarter ones: How to extract meaningful data from machine learning classifiers. *International Journal of Security and Networks*, 10(3):137–150.

Ganju, K., Wang, Q., Yang, W., Gunter, C. A., and Borisov, N. (2018). Property inference attacks on fully connected neural networks using permutation invariant representations. In *Proceedings of the 2018 ACM SIGSAC conference on computer and communications security*, pages 619–633.

Grari, V., Ruf, B., Lamprier, S., and Detyniecki, M. (2020). Achieving fairness with decision trees: An adversarial approach. *Data Science and Engineering*, 5(2):99–110.

Hartmann, V., Meynent, L., Peyrard, M., Dimitriadis, D., Tople, S., and West, R. (2023). Distribution inference risks: Identifying and mitigating sources of leakage. In *2023 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, pages 136–149. IEEE.

Huang, G. B., Mattar, M., Berg, T., and Learned-Miller, E. (2008). Labeled faces in the wild: A database forstudying face recognition in unconstrained environments. In *Workshop on faces in'Real-Life'Images: detection, alignment, and recognition*.

Kohavi, R. et al. (1996). Scaling up the accuracy of naive-bayes classifiers: A decision-tree hybrid. In *Kdd*, volume 96, pages 202–207.

Li, Z. and Zhang, Y. (2021). Membership leakage in label-only exposures. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, pages 880–895.

Liu, Y., Wen, R., He, X., Salem, A., Zhang, Z., Backes, M., Cristofaro, E. D., Fritz, M., and Zhang, Y. (2022). ML-Doctor: Holistic risk assessment of inference attacks against machine learning models. In *USENIX Security*.

Nasr, M., Shokri, R., and Houmansadr, A. (2018). Machine Learning with Membership Privacy using Adversarial Regularization. In *CCS*.

Shokri, R., Stronati, M., Song, C., and Shmatikov, V. (2017). Membership inference attacks against machine learning models. In *2017 IEEE symposium on security and privacy (SP)*, pages 3–18. IEEE.

Song, L. and Mittal, P. (2021). Systematic evaluation of privacy risks of machine learning models. In *USENIX Security*.

Stock, J., Wettlaufer, J., Demmler, D., and Federrath, H. (2023). Lessons learned: Defending against property inference attacks. In di Vimercati, S. D. C. and Samarati, P., editors, *Proceedings of the 20th International Conference on Security and Cryptography (SECRYPT)*, pages 312–323. SCITEPRESS.

Suri, A. and Evans, D. (2022). Formalizing and estimating distribution inference risks. *Proceedings on Privacy Enhancing Technologies*.

Tang, X., Mahloujifar, S., Song, L., Shejwalkar, V., Nasr, M., Houmansadr, A., and Mittal, P. (2021). Mitigating membership inference attacks by self-distillation through a novel ensemble architecture. *USENIX Sec.*

Tramèr, F., Zhang, F., Juels, A., Reiter, M. K., and Ristenpart, T. (2016). Stealing machine learning models via prediction {APIs}. In *25th USENIX security symposium (USENIX Security 16)*, pages 601–618.

Xu, L., Skoularidou, M., Cuesta-Infante, A., and Veeramachaneni, K. (2019). Modeling tabular data using conditional gan. In *Advances in Neural Information Processing Systems*.

Xu, R., Baracaldo, N., and Joshi, J. (2021). Privacy-preserving machine learning: Methods, challenges and directions. *arXiv preprint arXiv:2108.04417*.

Zhang, W., Tople, S., and Ohrimenko, O. (2021). Leakage of dataset properties in {Multi-Party} machine learning. In *30th USENIX security symposium (USENIX Security 21)*, pages 2687–2704.

Zhang, Z., Song, Y., and Qi, H. (2017). Age progression/regression by conditional adversarial autoencoder. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5810–5818.

Zhou, J., Chen, Y., Shen, C., and Zhang, Y. (2022). Property inference attacks against gans. In *30th Network and Distributed System Security Symposium (NDSS 2022)*.