Project on Responsible AI

Killian Heim, Oriane Longeron

Introduction

The US Adult Income dataset, sourced from Kaggle, was originally extracted by Barry Becker from the 1994 US Census Database. It contains anonymous data about various social and economic factors, including occupation, age, native country, race, capital gain, capital loss, education, work class, and more.

Each entry in the dataset is labeled based on income, categorizing individuals as earning either ">50K" or "≤ 50K" annually. This classification allows for the analysis of how different social factors correlate with income levels.

The dataset is divided into two CSV files:

- *adult-training.txt* : Contains data used for training models.
- *adult-test.txt* : Contains data used for testing models.

This dataset is commonly utilized for machine learning tasks focused on income prediction and social factor analysis.

In this project, we focus on analyzing the robustness of a Multi-Layer Perceptron (MLP) classifier. By training the MLP model on this dataset, we aim to examine how well the model predicts income levels based on these factors. At the same time, we investigate the model's vulnerability to adversarial and privacy attacks, with the objective of evaluating how robust the classifier is against these threats. Furthermore, we assess the presence of bias in the model, identify the sources of that bias, and explore strategies for mitigating it.

Adversarial attack and defense

This part of the project aims to assess the robustness of our MLP classifier against adversarial attacks and explore an effective defense strategy. To achieve this, we tested two attack methods: Fast Gradient Method (FGM) and Projected Gradient Descent (PGD), before implementing adversarial training to strengthen the model.

Initially, we used FGM, a fast attack that applies a single-step perturbation based on the gradient of the loss function. Its main advantage is its speed, making it an effective first approach to testing a model’s vulnerability. However, since it does not account for the non-linearity of the model’s decision boundary, it may overlook

certain exploitable weaknesses. This method was chosen because it allows us to quickly identify significant accuracy degradations. We tested multiple epsilon values, observing a sharp drop in accuracy until a saturation point where further increasing ε no longer caused major degradation.

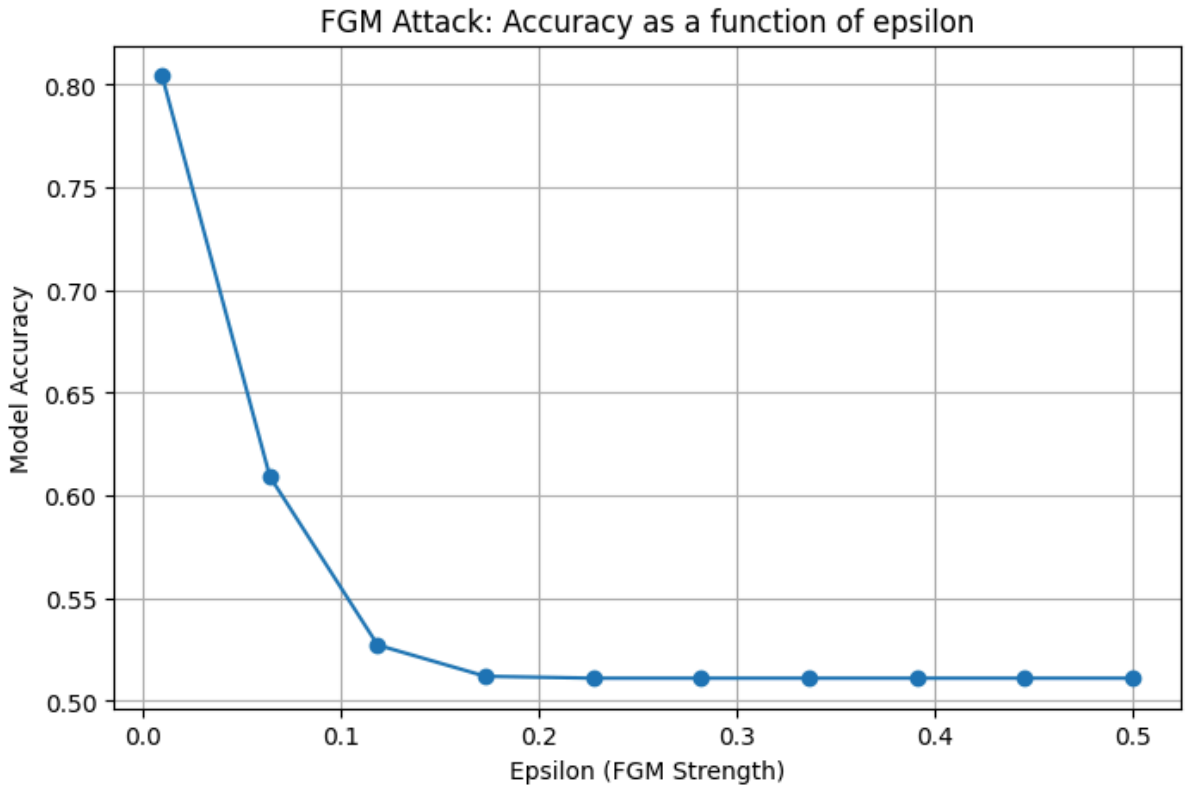

Figure 1 : *FGM Attack: Accuracy as a function of epsilon*

Next, we used PGD, a more powerful variant of FGM that applies multiple iterations of perturbations, recalculating the gradient at each step. This enables it to exploit more intricate weaknesses in the model, making it a significantly stronger attack. The main drawback of PGD is its higher computational cost compared to FGM (357.5s instead of 4.5s), but it remains a realistic threat against deep learning models. After applying the PGD attack, the model's accuracy significantly dropped to 51% with ε = 0.06, further illustrating its vulnerability to such sophisticated adversarial manipulations. This emphasizes the necessity for robust defense strategies to strengthen the model’s resilience against these advanced attacks.

To counter these attacks, we implemented adversarial training, which involves exposing the model to adversarial examples during its learning process. This approach enhances its ability to handle such perturbations and maintain higher accuracy on attacked data. We opted to augment the training dataset with FGM-perturbed examples (ε = 0.2). This value was chosen as it offers a balance between significantly impacting robustness and keeping the perturbation reasonable.

After training, the model's accuracy against adversarial examples improved significantly, increasing from 51% to 81%.

The conducted experiments highlighted the model's initial vulnerability to adversarial perturbations. While FGM is fast and simple, it is limited in its ability to deceive the model, whereas PGD is more effective but computationally expensive. Adversarial training proved to be a relevant solution, significantly enhancing the model's robustness while maintaining good performance on unmodified data. This approach thus provides an effective trade-off between performance and protection against adversarial attacks.

Privacy attack and defense

In this part of the project, we explored privacy risks in machine learning models by performing a Membership Inference Attack (MIA). This type of attack tries to determine whether a given data sample was used to train a model, which can reveal sensitive information. To defend against this, we applied Differentially Private Stochastic Gradient Descent (DP-SGD), a training method that adds noise to the learning process to protect individual samples. Our goal was to analyze how effective this defense is in reducing the success of the attack while maintaining the model's accuracy.

Presentation of MIA

We distinguish 3 different models in a MIA :

- The target model on which the attack is performed
- The shadow models that are meant to imitate the target model
- The attack model which predicts if a data is present in the training data

The following figure, extracted from the thesis *Membership Inference Attacks on Machine Learning Models: Analysis and Mitigation* [1] illustrates how these models interact during a MIA :

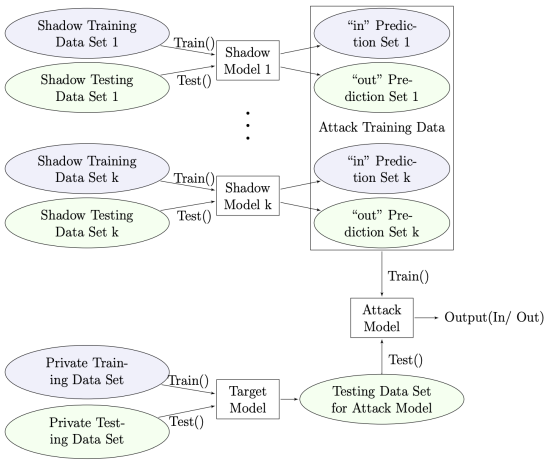

Figure 2 : *Membership Inference Attack on ML models*

During a Membership Inference Attack (MIA), the attacker creates several shadow models that are designed to solve the same machine learning problem as the target model. To train these shadow models, the attacker selects ***k*** datasets that are similar to the original dataset used to train the target model. As with standard machine learning training, a portion of each dataset is used for training the shadow models, while the remaining data is used for testing.

The predictions made by our shadow models on both their training and test sets are then fed into the attack model. In our case, each prediction is represented as a probability ***P***, which corresponds to the confidence score of the binary classification task.

Our target ***t*** for the attack model is also binary :

- ***t = 0*** if the sample was not present in the trainset of the shadow models (present in the testset)
- ***t = 1*** if the sample was present in the trainset of the target models

The attack model can infer whether a data point was used in training because samples from the training set tend to produce more confident or distinct probability distributions, whereas unseen samples (from the testset) generally result in more uncertain or evenly spread probabilities. This difference allows the attack model to distinguish between samples that were part of the training set and those that were not.

Presentation of DP-SGD

Differential Private Stochastic Gradient Descent (DP-SGD) is a modification of the standard SDG algorithm used to protect the privacy of the trainset samples of a model. The algorithm achieves this by reducing the impact of a sample on the training of the model, making it harder for the attacker to infer whether a specific sample was part of the training data or not. DP-SGD is applied to the target model as an optimization algorithm during the training phase. It is based on two methods:

- **Gradient clipping:** Gradient clipping limits the size of each training sample's gradient before updating the model. This ensures that each sample does not have too much impact on the learning process and prevents the model from memorizing specific samples.
- **Noise addition:** Noise is then added to the clipped gradients. In practice, this parameter is controlled by a *noise multiplier.* If the level of noise is high, the model's outputs will be more random, increasing privacy protection. However, this process also alters the accuracy of the target model.

The following figure extracted from the scientific article [2] provides an illustration for DP-SGD :

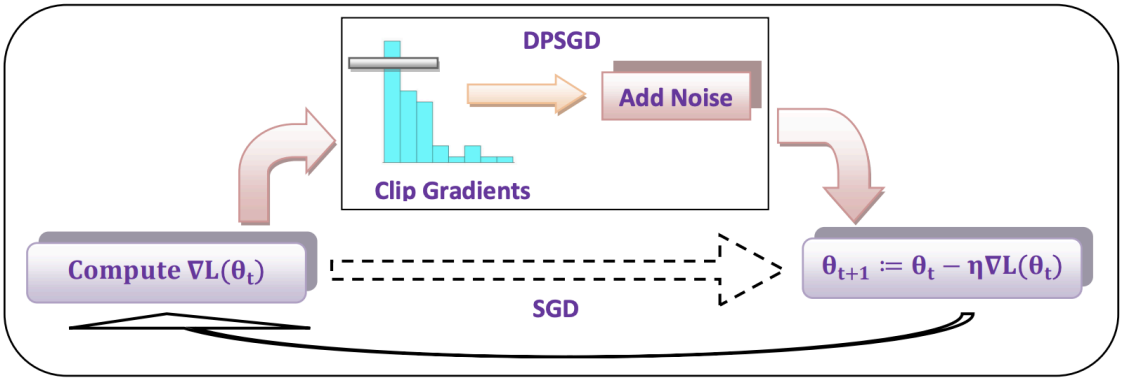

Figure 3 : *Illustration of the DP-SGD algorithm*

It is important to tune these parameters correctly in order to assure a protection of the training data without altering the performance of the target model.

Experimental results

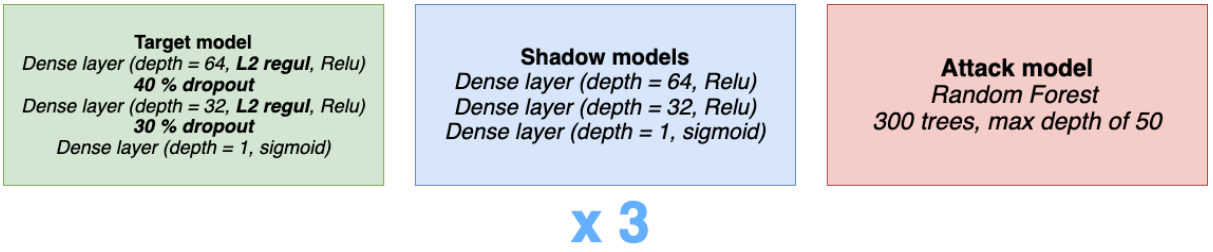

Figure 4 : *Models used for the experimental MIA*

For our Membership Inference Attack (MIA), we used a MLP as the target model, which includes dropouts and L2 regularization to reduce overfitting. These techniques help prevent the model from memorizing specific training data, improving generalization. For the shadow models, we used a simpler MLP architecture, assuming that the attacker does not have access to the full details of the target model and only has a simplified version. Finally, for the attack model, we used a Random Forest, which is effective in classifying whether a sample was part of the target model's training set based on the predictions from the shadow models. This setup allows us to evaluate the effectiveness of the MIA against a model trained with and without differential privacy defenses.
The training data for both the target model and the shadow models were drawn from the same dataset, but with different random seeds for the data split. This reflects the scenario where the attacker uses a dataset similar to the one used for training the target model.
For our defense, we used the tensorflow_privacy package that provides several ways of implementing defenses to MIA. Particularly, we used the function

DPKerasSGDOptimizer that allows us to apply differential privacy during the training of the target model.
We can observe the effect of differential privacy on the accuracy of the target model by plotting the accuracy's evolution at each epoch during the training.

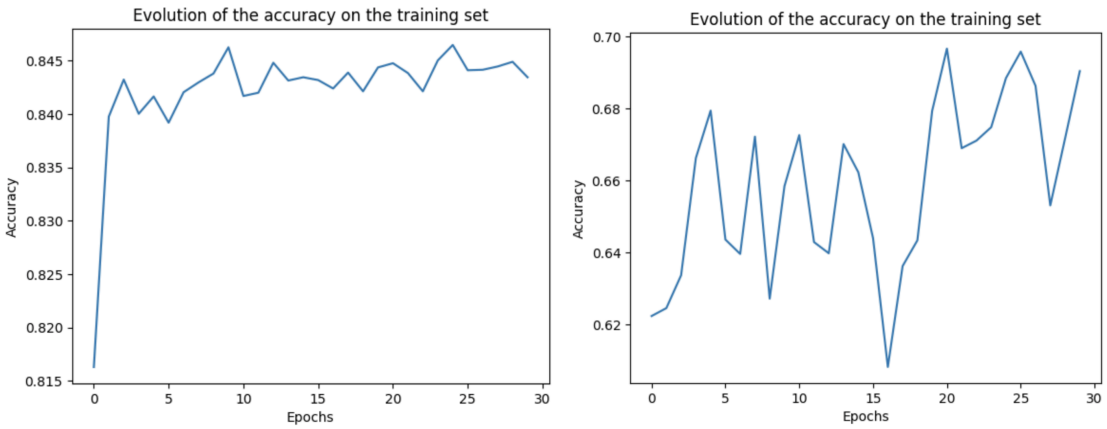

Figure 5 & 6 : *Evolution of accuracy without (left) and with (right) differential privacy*

With DP-SGD, the model struggles to converge during the first epochs and the accuracy fluctuates more compared to the standard training. This behavior is expected due to the added gradient clipping and noise. Although DP-SGD leads to lower performances of the model, we were able to reach an accuracy of nearly 70% on the trainset which is correct compared to the 84% obtained with the standard model.
On the testset, we obtained an accuracy of 85% for the standard model and an accuracy of 75% after application of DP-SGD.
Shadow models have similar accuracies that the standard model which is coherent since their architectures are close.
To evaluate the performance of the MIA, we computed two scores :

- The accuracy of the attack on the shadow models data
- The accuracy of the attack on the target model's data

Only the first score would be accessible to the attacker since in practice it is not possible to access the probabilities computed by the target model. The second indicator helps us to know if the DP-SGD is efficient against the MIA.

	Target Model	Shadow models
Without Defense	0.624182	0.615936
With Defense	0.448113	0.619314

Figure 7 : *Accuracy of the MIA on the target model and the shadow models*

We obtained an accuracy of approximately 62% for the MIA on the target and shadow models in the case where the target model is not protected. We observe that the accuracy is slightly better on the target model which might seem abnormal. However, this difference is not significant and is explainable by the fact that the models were trained on similar data.
We also observe that the accuracy is over 60% in this case. This means that the attack model is able to leak a fraction of the training data. Indeed, since the attack model is performing a binary classification, an accuracy of 50% would mean that the predictions are not better than randomness.
During our experiments we were also able to make the MIA more efficient by making the target and shadow models overfit to their training data.

When DP-SGD is applied to the target model, we witness a reduction of the attack accuracy to 45%. The fact that the accuracy is near to 50% shows that the attack is no longer efficient. In this case, the predictions made by the attack model are close to randomness.

The conducted experiments highlighted how Membership Inference Attack (MIA) can be used to access the training data of a model. The performance of a MIA is variable and further experiments on the architecture of the attack model could influence the attack accuracy. The implementation of DP-SGD was efficient for protecting the training data of the target model. However, the hyperparameters of DP-SGD can drastically affect the performance of the target model. Techniques such as cross-validation could be explored to find suitable hyperparameters that allow an efficient defense and that limit the decrease of accuracy.

Model bias and fairness

Bias in machine learning happens when a model makes unfair or inaccurate predictions, often favoring one group over another. This can happen if the training data is not balanced or if the model learns patterns that reflect real-world unfairness.

In our dataset, we have notices different attributes that can lead to bias for the predictions of our MLP model :

- ***Race:*** Risk of racial discrimination in economic decisions
- ***Gender:*** Pay differentials and gender stereotypes can introduce bias
- ***Native country:*** Can be a proxy for socio-economic disparities based on origin.
- ***Age:*** Can lead to discrimination linked to professional seniority or access to economic opportunities
- ***Marital status:*** May be correlated with income inequalities based on societal norms

To better understand and address bias in our model, we need to analyze how these attributes impact predictions. Some demographic groups may receive systematically different levels of income due to imbalances in the training data, historical discrimination, or societal inequalities reflected in the dataset. Identifying these biases is crucial for ensuring fairness in our model.

To assess fairness, we opted for the Fairlearn library instead of Justicia due to compatibility issues. Several fairness metrics were used, including Statistical Parity (SP), Equality of Opportunity (EO), and Predictive Value Parity (PVP). Statistical Parity ensures that different demographic groups have the same likelihood of receiving a positive outcome, while Equality of Opportunity evaluates whether individuals who deserve a positive outcome are treated fairly across groups. We analyzed fairness across different sensitive attributes, including gender, race, and marital status, both individually and in combination.

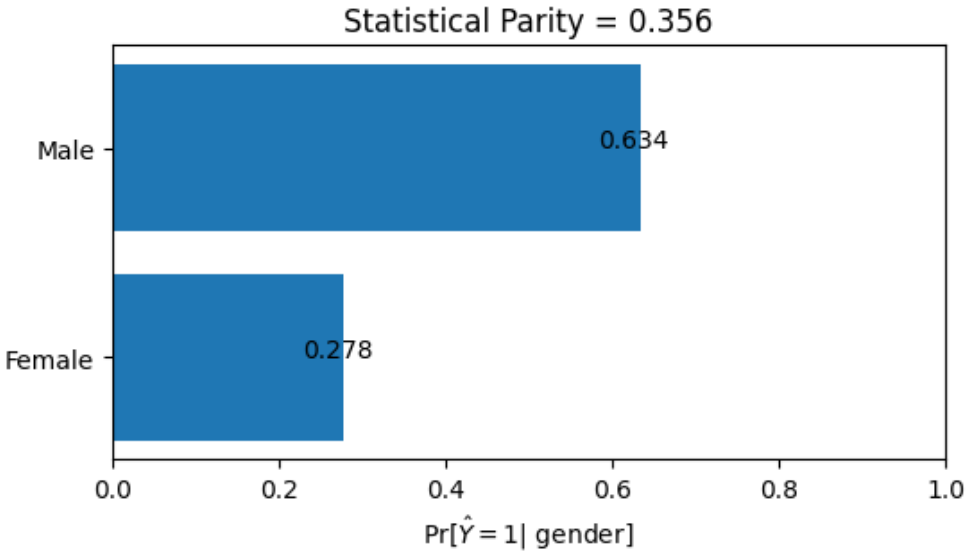

Figure 8 : Statistical Parity on Gender.

Results revealed significant disparities in the model's predictions. When considering gender, males had a higher probability of being classified as having a high income compared to females, indicating gender-based bias. The disparity was even more pronounced when race was introduced, with White and Asian-Pac-Islander individuals having a significantly higher probability of receiving a positive income classification compared to Amer-Indian-Eskimo and Other racial groups. The combination of race and gender further exacerbated the disparities, showing that White and Asian-Pac-Islander males were the most advantaged, while Black and Amer-Indian-Eskimo females were the most disadvantaged. The high values of SPD and EOD confirmed both allocative harm and representational harm in the model’s predictions.

We attempted to explain the sources of unfairness in our model using FairXplainer, a tool designed to provide insights into how different features contribute to biased predictions. However, we faced compatibility issues that made its implementation challenging, and we were unable to display the explanation.

To mitigate these fairness issues, we applied a pre-processing fairness algorithm, specifically Reweighing, which assigns different instance weights to privileged and unprivileged groups. This technique aims to correct for bias before training by adjusting the importance of different groups in the dataset. A new MLP model was then trained on the transformed dataset, incorporating these fairness adjustments. The retrained model maintained a similar level of accuracy while reducing bias. Comparing fairness metrics before and after applying Reweighing, we observed a significant reduction in Statistical Parity Difference (0.243 instead of 0.356), indicating improved fairness in the model's predictions. Other strategies such as in-processing approaches (e.g., adversarial debiasing) or post-processing corrections (e.g., equalized odds adjustments) could also be explored to further enhance fairness. While improvements were achieved, ongoing monitoring and refinement remain essential to ensure that fairness considerations are continuously addressed in real-world applications.

Conclusion

In conclusion, this project explored the robustness, privacy, and fairness of an MLP classifier using the US Adult Income dataset. We demonstrated that the model is vulnerable to adversarial attacks, including the Fast Gradient Method (FGM) and the Projected Gradient Descent (PGD) but adversarial training can be used as a defense. Applying DP-SGD showed the compromise between privacy and model performance, and how it can protect training data from Membership Inference Attacks (MIA). Our fairness analysis also revealed significant biases in the model's predictions, highlighting the need to address these issues. Reweighing helped reduce these biases, and further research into other fairness techniques could improve this even more. Ultimately, this project emphasizes the importance of considering robustness, privacy, and fairness when developing machine learning models, to ensure AI is used responsibly.

References

[1] - **Membership Inference Attacks on Machine Learning Models: Analysis and Mitigation** *by Md Shamimur Rahman Shuvo*

[2] - **Membership Inference Attack against Differentially Private Deep Learning Model** *by Md Atiqur Rahman, Tanzila Rahman, Robert Laganière , Noman Mohammed, Yang Wang, University of Ottawa*