Reproducibility Challenge 2022: FR-Train: A Mutual Information-Based Approach to Fair and Robust Training

Manas Kale 261000867

Gauri Sharma 261026894

Avinash Bhat 260969537

Reproducibility Summary

Scope of Reproducibility

To confirm the claim that the FR-Train algorithm[1] yields better accuracy and fairness (measured by disparate impact) in comparison to other fairness algorithms like Fairness Constraints, Label Bias Correction and Adversarial Debiasing. We reproduce three out of five experiments performed in the paper. We perform the experiments on the synthetic dataset and observe that the FR-Train algorithm indeed provides us a high metric for fairness (as measured by Disparate Impact) and also accuracy. We performed ablation studies and confirmed the claim that the algorithm achieve both model fairness and training robustness using different discriminator networks.

Methodology

The reproducibility was performed on two different laptops with Intel i7 processors(16 GB of RAM). The author's code was reused with certain modifications to account for newer library versions. We had to make modifications to account for newer PyTorch versions and to allow the pipeline to work with AdultCensus and COMPAS datasets.

Results

We were able to exactly reproduce the results for synthetic dataset as reported in the paper. Due to high computational requirements for the COMPAS and AdultCensus datasets, we were not able to replicate all of the experiments conducted and presented in the paper. Out of the three experiments that we were able to reproduce, two of them were within 1% of the reported value which supports the papers conclusions.

What was easy

The paper was understandable and it was quite fascinating to follow its structure. Along with the theoretical concept, the mathematical equations provided ease to reformulate the paper. It was easy to follow the author's code, especially the model architecture portion. The code is well commented, readable and organized in an efficient manner. This code can be easily adopted to use with other datasets.

What was difficult

The authors did not provide any data preprocessing functions for COMPAS and AdultCensus datasets. Computational resources required for training on these datasets were also not mentioned. Also, an older version of PyTorch was used, which caused incompatibility issues with the latest stable version.

Communication with original authors

We emailed the authors our preliminary results and noted the modifications required to make their code work with COMPAS and AdultCensus datasets. We did not receive any response at the time of submission.

Submitted to ML Reproducibility Challenge . Do not distribute.

1 Introduction

Training of the machine learning models has to be robust to data bias, noise and any poisoning of data. Most models use public datasets for training, which can contain subjective or poisoned data, which could be due to nature of the data, its collection methods or sometimes due to sophisticated data poisoning attacks. Authors claim that the existing model fairness training techniques cause a performance degradation because they treat the poisoned data as bias. Model training while addressing fairness involves a trade off with accuracy. One of the premise to the paper is that not addressing data poisoning prior to solving model fairness can lead to a worse trade off. Sanitizing the data to address data poisoning is difficult without having any knowledge of the model.

The authors therefore propose an adversarial training based fairness and robustness training method called FR-Train, which is an extension of an existing state of the art method called Adversarial Debiasing, and adds a robustness discriminator that refers a clean validation data and ensures that the predictions are consistent with that of the clean data and is used to further improve the fairness training through re-weighting examples. Authors claim that the algorithm is robust to data poisoning and can be adjusted to maintain reasonable accuracy and fairness even if the validation set is too small or unavailable.

The paper also demonstrates how a clean validation set can be constructed through crowdsourcing and releases two new datasets built using Amazon Mechanical Turk.

2 Scope of reproducibility

The primary claim made by the paper is that the FR-Train algorithm yields better accuracy and fairness (measured by disparate impact) in comparison to other fairness algorithms like Fairness Constraints[2], Label Bias Correction[3] and Adversarial Debiasing[4]. This claim is backed by experiments on synthetic and real data.

The next claim in the paper is that a 5% validation set is sufficient for the FR-Train algorithm to maintain the accuracy and fairness obtained from clean data. This is backed by experiments on the validation set and later the crowdsourced *clean* dataset.

The paper also claims that the FR-Train algorithm can achieve **both** model fariness and training robustness as opposed to the prior algorithms that can achieve only one of those. This claim is backed by the ablation study.

Detailed list of experiments is as follows.

1. **Synthetic data experiment**: A synthetic dataset with 2000 data points is constructed and later poisoned and is used for simulating and testing the effectiveness of the algorithm. We replicate this experiment and observe similar results to the paper.
2. **Validation set requirement**: The size of validation set (as a percent of the actual dataset) that is required by the FR-Train algorithm for maintaining the accuracy and fairness obtained on the clean data.
3. **Real data experiments**: A clean dataset is constructed using crowdsourcing and is used as the validation set on which the FR-Train algorithm is run.
4. **Ablation studies**: Experiments to investigate the effect of each component of FR-Train. We were able to reproduce the results.
5. **Error range of FR-Train**: The experiments on the poisoned data are re-conducted with ten different random seeds and the mean and standard deviation are reported. We are able to reproduce this experiment.

The results obtained for each of the experiments are included in the Result section. Due to time and computational constraints, we were not able to fully reproduce large-scale examples for COMPAS and AdultCensus datasets.

3 Methodology

Since the code is publicly available on GitHub, we decided to reuse the code. Authors provide the synthetic data and the crowdsourced data. However the two datasets that are used (COMPAS and AdultCensus) are not provided in the repository but are publicly available and can be easily sourced as well. The code is well documented, and are accompanied by two Jupyter notebooks that simulate the architecture for both clean and poisoned data.We had some difficulty running it with Pytorch version 1.11.0 due to some code having been deprecated. To fix this, we changed Pytorch API calls and some dataframe shapes. The updated code is available in the forked codebase available **here**.

3.1 Model descriptions

The model follows a GAN architecture and consists of two discriminators (one each for fairness and robustness) and one generator. The generator is a neural network with zero or one hidden layer. The generator uses Adam optimization

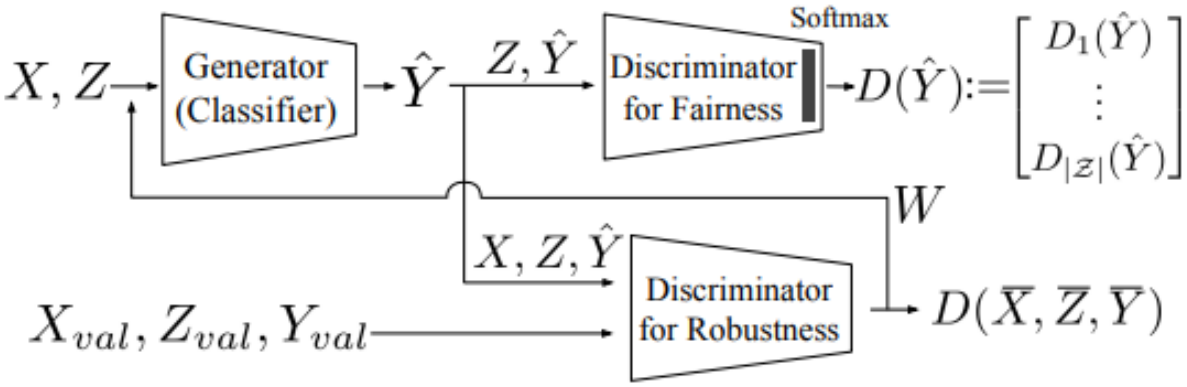

Figure 1: FR-Train Architecture

algorithm. The discriminator for fairness is a single layer neural network. Discriminator for robustness is a neural network with one hidden layer with 8 or 16 nodes. Discriminators use stochastic gradient descent algorithm.

According to the paper, the hyperparameters of the training discriminator were frozen for the first few epochs until the generator achieves a certain accuracy for stabilizing the training procedure. We followed this process in our training as well. The generator to discriminator update ratio of 1:3 was observed for the training.

3.2 Datasets

The experiments make use of a synthetic dataset, two real world datasets and validation datasets for the real world datasets. The synthetic dataset consists of 2000 data points with three attributes of which one is sensitive. The sensitive attribute is binary in nature. The non sensitive attributes have a binary distribution while the sensitive attribute has a Bernoulli distribution. This synthetic dataset is poisoned using label flipping which maximises the performance degradation. This dataset is available in the GitHub repository associated with the paper.

The real world datasets are

1. **COMPAS**: Data containing criminal history, jail and prison time, demographics and COMPAS risk scores for defendants from Broward County. It is publicly available.
2. **AdultCensus**: Predict whether income exceeds 50k USD per year based on census data. It is publicly available.

These datasets are poisoned by flipping the labels with z = 1 so as to maximize the accuracy performance degradation.

Finally, a *clean* validation dataset is created for the real world datasets using the Amazon Mechanical Turk platform[1], where a human *worker* looks at various attributes and provides a label to the data. For the AdultCensus dataset, a worker predicts if a person has an income of at least $50K. For the COMPAS dataset, a worker predicts if a criminal will reoffend in two years. These crowdsourced datasets are available in the GitHub repository associated with the paper.

The data is preprocessed in the following manner. First, example reweighting is performed on the data which generated weights for the training examples in each combination of group and label differently to ensure fairness before classification. Then a probabilitic transformation model is used to identify the fairness, individual distortions and data fidelity constraints. Then a latent representation is found that encodes the data and obfuscates the information about protected attributes. Finally a disparate impact remover edits the feature values to increase group fairness.

[1] https://www.mturk.com/

3.3 Hyperparameters

There are three hyperparameters that are used:

1. **C**: The threshold on the loss ratio that is used for re-weighting examples for fairness training. This value was set to be between 0 and 3.
2. λ_1: Hyperparameter for fairness training.
3. λ_2: Hyperparameter for robustness training. For training with clean data, λ_1 is set to 0.1 and λ_2 was varied from 0 to 0.85. For training with poisoned data, λ_2 was set from 0.2, 0.3 and 0.35 and λ_1 was varied from 0 to 0.95 - λ_2. We followed these specifications as per the paper. We performed manual search over the hyperparameters. The best outcome is highlighted in the table.

We studied the effects of these hyperparameters in the ablation study.

3.4 Experimental setup and code

The baseline metrics used in the paper are extended versions of fairness algorithms Fairness Constraints, Label Bias Correction and Adversarial Debiasing with meta learning for robustness. We do not perform the baseline experiments since they are used for comparison only and assume that the values mentioned in the paper are accurate.

Experiment results are evaluated using a fairness measure called disparate impact. Disparate Impact (DI) is measured as $min\left\{\frac{P(\hat{Y}=1|Z=z_1)}{P(\hat{Y}=1|Z=z_2)}, \frac{P(\hat{Y}=1|Z=z_2)}{P(\hat{Y}=1|Z=z_1)}\right\}$ where z_1 and z_2 are sensitive attributes. Apart from disparate impact, the accuracy values are also shown.

The code used for reproducing the experiments are available in this GitHub repository. A notebook with plots is also included with the submission.

3.5 Computational requirements

All the reproducability experiments were run on the CPU. The average run time of each experiment is around 20 seconds for the synthetic dataset. This is due to the fact that the datapoints are mostly numeric and relatively small with a size of 2000 instances. However, it is much larger for COMPAS and AdultCensus datasets. The experiments were run on two laptops, both with Intel i7 CPU with 16 GB RAM.

4 Results

4.1 Results reproducing original paper

The FR-Train paper claims to have the following results for synthetic dataset.

Dataset	**Synthetic**	
Data	**Disparate Impact**	**Accuracy**
Clean Data	.818	.807
Poisoned Data	.827	.814

The specific hyperparameters used to obtain these results (λ_1, λ_2) were not explicitly mentioned in the paper. It seems the best hyperparameters for the synthetic poisoned dataset were left in the release version of the code.

4.1.1 Result: Synthetic Dataset

Since the best hyperparameters for poisoned dataset were left in the code, we were able to exactly reproduce DI and accuracy values for synthetic poisoned data. For the clean poisoned data, we tried various hyperparameters and were able to find best result of 0.754 DI and 0.822 accuracy. See Figures 2 and 3 for loss plots with the best hyperparameters.

4.2 Results beyond original paper

Since this paper claims using two discriminators (one for fairness and other for robustness) is important, it made most sense to perform an ablation study that disables these two components. Although the authors did perform an ablation study, they did not use the synthetic dataset. So we used the synthetic dataset for our ablation test.

4.2.1 Additional Result

We performed two ablation studies - one that disabled the fairness discriminator ($\lambda_1 = 0$) and other that disabled the robustness discriminator ($\lambda_2 = 0$). Since these parameters are the weights for these network's respective losses in the objective function, setting them to 0 effectively disables these networks. We performed this test for both clean and poisoned data. We expected this to make no different on the cleaned data as it had no inherient bias anyway, so the fairness/robustness networks should not have any effect. We expected a loss in performance for the poisoned dataset since the fairness/robustness networks are designed to combat poison. Our hypothesis was verified by looking at figures 4, 5, 6 and 7 - we observe that generator loss does not change much for poisoned datasets with fairness disabled. Also, we got an accuracy of 0.86 on clean and 0.772 on poisoned data.

5 Discussion

We observe that the results obtained from reproducing closely match with that reported in the paper. All hyperparameters were not included in the paper, but our results show the author's claims are fairly accurate. We were able to verify all the claims on synthetic data and found no major discrepencies in the authors claims.

The primary claim that FR-Train yields better accuracy and fairness can only be verified by running the other algorithms on same datasets. Since the scope of this project did not require this, we were only able to verify the performance of FR-Train.

The next claim that a 5% validation set is sufficient for the FR-Train algorithm to maintain accuracy and fairness obtained from clean data was also verified through the tests we performed on synthetic dataset.

The last claim that the FR-Train algorithm can achieve **both** model fariness and training robustness as opposed to the prior algorithms that can achieve only one of those was also verified through our ablation study. Due to both time and computational constraints, we were not able to reproduce all the large-scale experiments in the paper for the real datasets, which served as a roadblock in doing a complete assessment of the claims presented in the paper.

The error ranges we obtain are slightly higher than what the paper reports, but we still think that they are within an acceptable range.

5.1 What was easy

We found that the code was easy to run and was well documented and organized following SOLID principles.

5.2 What was difficult

All the results included in the paper were not included in the code. The authors could have provided hyperparameter configuration files for further reproducibility. Working with the real world dataset proved to be challenging. We did not construct the clean validation set since the process involved crowdsourcing using Amazon Mechanical Turk, and was a larger project to undertake. Due to both time and computational constraints, we were not able to reproduce all the large-scale experiments in the paper for the real datasets, which served as a roadblock in doing a complete assessment of the claims presented in the paper. Moreover, the authors didn't provide the code for both of these datasets.

5.3 Communication with original authors

We tried communicating with the authors but have not received a response from them at the time of submitting this report.

References

[1] Y. Roh, K. Lee, S. E. Whang, and C. Suh, "Fr-train: A mutual information-based approach to fair and robust training," in *Proceedings of the 37th International Conference on Machine Learning*, ser. ICML'20, JMLR.org, 2020.

[2] M. B. Zafar, I. Valera, M. Gomez-Rodriguez, and K. P. Gummadi, "Fairness constraints: Mechanisms for fair classification," in *AISTATS*, 2017.

[3] H. Jiang and O. Nachum, "Identifying and correcting label bias in machine learning," in *AISTATS*, 2020.

[4] J. Zhang and E. Bareinboim, "Fairness in decision-making — the causal explanation formula," in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, ser. AAAI'18/IAAI'18/EAAI'18, New Orleans, Louisiana, USA: AAAI Press, 2018, ISBN: 978-1-57735-800-8.

Appendix

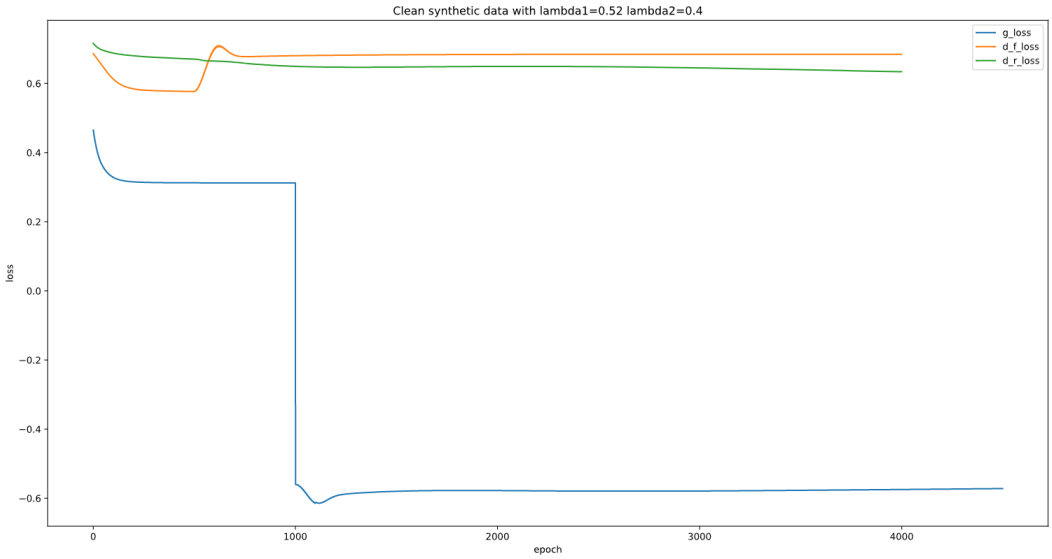

Figure 2: Loss plots for clean synthetic data with best hyperparameters

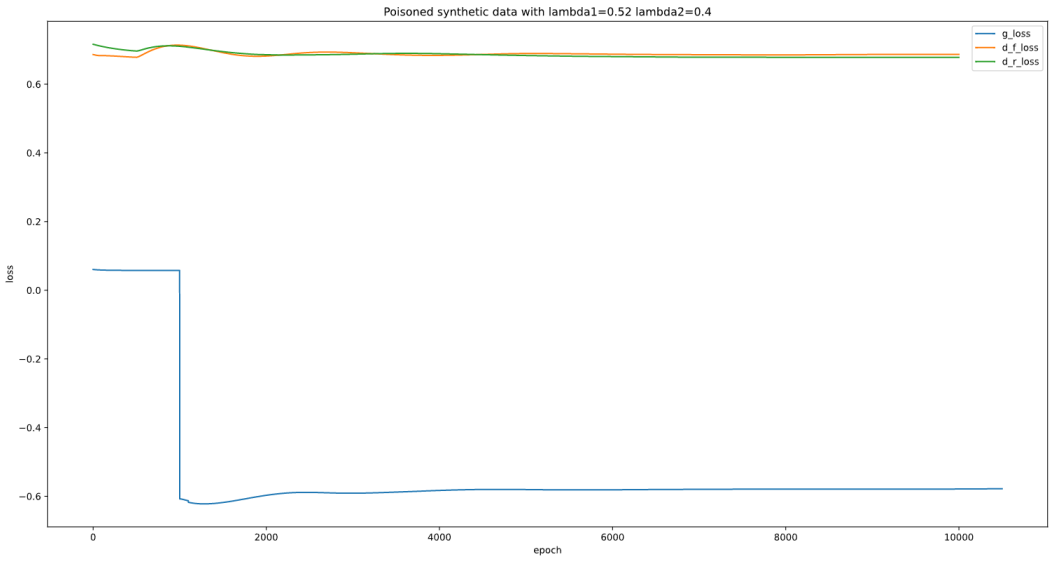

Figure 3: Loss plots for poisoned synthetic data with best hyperparameters

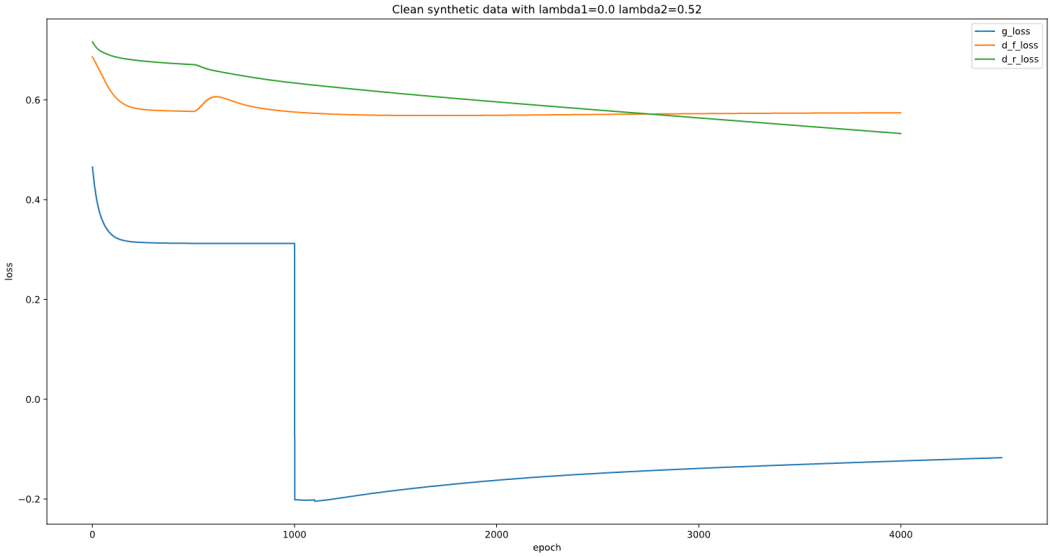

Figure 4: Loss plots for clean synthetic data with fairness disabled

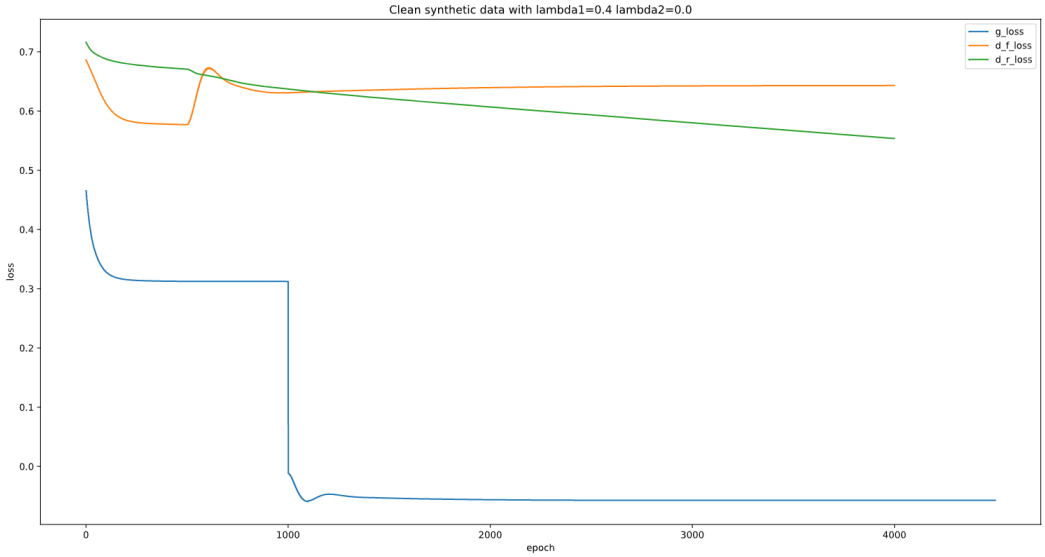

Figure 5: Loss plots for clean synthetic data with robustness disabled

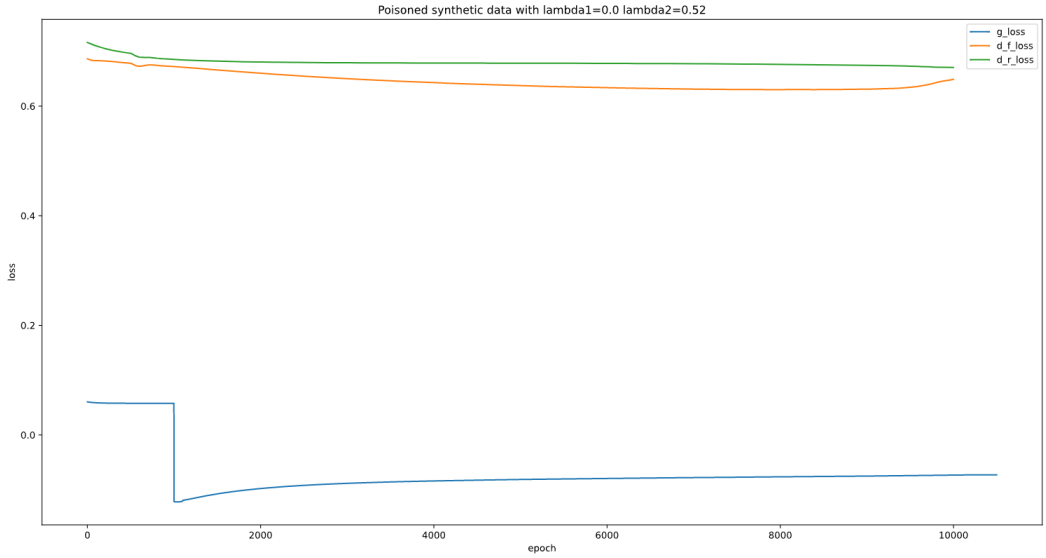

Figure 6: Loss plots for poisoned synthetic data with fairness disabled

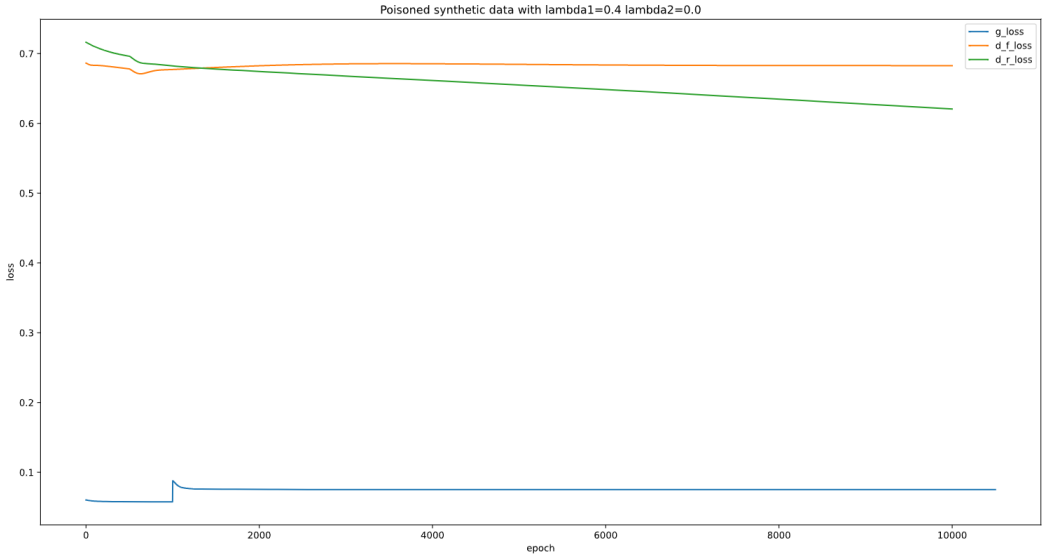

Figure 7: Loss plots for poisoned synthetic data with fairness disabled