Disparate Impact on Group Accuracy of Linearization for Private Inference

Saswat Das [*1] **Marco Romanelli** [*2] **Ferdinando Fioretto** [1]

Abstract

Ensuring privacy-preserving inference on cryptographically secure data is a well-known computational challenge. To alleviate the bottleneck of costly cryptographic computations in non-linear activations, recent methods have suggested linearizing a targeted portion of these activations in neural networks. This technique results in significantly reduced runtimes with often negligible impacts on accuracy. In this paper, we demonstrate that such computational benefits may lead to increased fairness costs. Specifically, we find that reducing the number of ReLU activations disproportionately decreases the accuracy for minority groups compared to majority groups. To explain these observations, we provide a mathematical interpretation under restricted assumptions about the nature of the decision boundary, while also showing the prevalence of this problem across widely used datasets and architectures. Finally, we show how a simple procedure altering the fine-tuning step for linearized models can serve as an effective mitigation strategy.

1. Introduction

Private Inference is the process of performing inference tasks on encrypted or private data, ensuring that the data remains confidential throughout the process. It has found application in ML settings where sensitive data is required for inference but should not be revealed to the model, for instance, when the model is owned by a cloud service provider and is not necessarily trusted. Although promising, cryptographic computations are notoriously computationally expensive when applied to nonlinear functions (Mishra et al., 2020), and, in the context of ML inference, Rectifier Linear Function (ReLU) activations are often identified as the primary cause of this computational burden (Cho et al., 2022; Jha et al., 2021; Kundu et al., 2023).

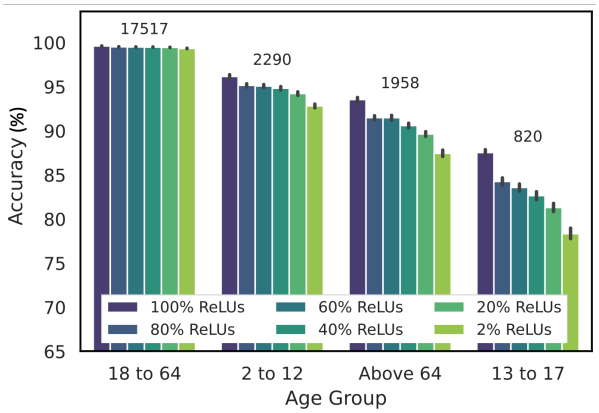


Figure 1. ResNet18 accuracy on UTKFaces across various ReLU linearization budgets using SNL (Cho et al., 2022). 100% ReLUs corresponds to the original model. Subgroup sample sizes are shown above the corresponding bars. The accuracy for the majority group remains almost unchanged while other groups, in particular the minority one, are diversely impacted.

To address this challenge, various approaches have proposed to approximate these non-linear activations using linear surrogates. The objective is to develop a new model that minimizes the number of ReLU activations while maintaining the highest possible accuracy. Over time, the proposed frameworks have evolved in sophistication—it has been demonstrated that not only the quantity of ReLU activations matters (Jha et al., 2021), but their placement within the network is also crucial (Cho et al., 2022). Recent methods have advanced to the point where they can manage both the ReLU budget and the distribution of linearized neurons across the network, resulting in faster inference times with minimal sacrifice in accuracy (Kundu et al., 2023).

This paper builds on this body of work and observes that while these methods are indeed effective at trading run-time reduction for marginal global accuracy losses, these accuracy decreases are unevenly distributed among different subgroups. We find that the accuracy loss is more pronounced for underrepresented subgroups and that this impact intensifies as more ReLUs are linearized. This effect is depicted in Figure 1, which shows the accuracy impact across age groups on a facial recognition task. Remarkably, while the majority group (left-most) remains relatively un-

[*]Equal contribution [1]University of Virginia, Charlottesville, VA, USA [2]New York University, New York, NY, USA. Correspondence to: Saswat Das <saswatdas@email.virginia.edu>, Marco Romanelli <mr6852@nyu.edu>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

harmed, minorities exhibit considerable reduction. *These results are far from "expected"*: they challenge the assumption that linearization uniformly affects the learning model and underscore the need for different approaches in model optimization, particularly in diverse datasets.

Contributions. This paper makes the following contributions: **(1)** It observes, for the first time, a Matthew effect on the reduction of precision caused by the targeted linearization process adopted to reduce the number of ReLU functions in private inference methods. Further, it shows that such disparity is exacerbated as the number of approximated ReLU functions grows. **(2)** Next, it presents a mathematical interpretation of this phenomenon, which relates these disparity effects with the approximation capabilities of ReLU functions when assumptions on the space of the decision boundaries can be made. **(3)** It further shows that the effect of ReLU reduction on fairness is vastly present for commonly deployed algorithms and models when trained on unbalanced datasets. **(4)** Finally, the paper proposes a simple yet effective mitigation strategy that can be applied to any framework for ReLU reduction during the fine-tuning phase, showing favorable results in terms of fairness and overall accuracy reduction.

2. Related Work

Frameworks for ML Private Inference. The relationship between presence of ReLU activations and the inference time of a private inference model was first investigated by Mishra et al. (2020). The seminal solution proposed in this work is based on automatic generation of neural network architecture configurations, that navigates the performance-accuracy trade-offs over a given cryptographic protocol. Building on this intuition, Ghodsi et al. (2020); Lou et al. (2021) propose to selectively replace ReLU functions with, computationally more efficient, polynomial operations. DeepReDuce (DR) (Jha et al., 2021) takes this approach a step further by proposing a multi-step optimization, where portions of the network are replaced with linear functions, and then finetuned to recover the accuracy of the original model. The main drawback of these methods is the limited control over the "distribution" of the ReLU throughout the net. To tackle this issue, Selective Network Linearization (SNL) (Cho et al., 2022) proposes a custom activation node, where the linear and rectified behavior are parametrized and learned during training through a parametric mask. In line with this paradigm, the approach in Kundu et al. (2023) introduces a novel measure of non-linearity sensitivity that helps reduce the need for manual efforts in identifying the best ReLU budget and placement.

Piece-wise Approximation with ReLU Functions. In addition to observing disparate impacts from methods that reduce the number of ReLU functions in neural networks, our work seeks to explain these effects by linking them to the expressiveness and approximation capabilities of a Deep Neural Network (DNN). These studies trace back to seminal works in Cybenko (1989); Hornik (1991). The widespread use of ReLU activations to introduce non-linearity in the learning of decision boundaries has motivated early studies about their approximation power (Montúfar et al., 2014; Pan & Srikumar, 2016). Several papers have observed the effect of ReLU activations in the learning process (Hanin & Rolnick, 2019; Lin & Jegelka, 2018; Lu et al., 2017; Shen et al., 2022; Yarotsky, 2017). Among others, Arora et al. (2018); Liu & Liang (2021) distinguish themselves for contributions to the field, analyzing the connection between the number of ReLU activations and that of the approximated piecewise functions, along with a ReLU-dependent bound on the approximation error.

Privacy and Fairness Tradeoff in Machine Learning. Among other works on the tradeoff between privacy and fairness in machine learning, Bagdasaryan et al. (2019) provides mostly empirical evidence of the disparate effect of differentially private training algorithms on fairness, while Farrand et al. (2020) further investigates this phenomenon by considering different levels of differential privacy and unbalance in the datasets. These works provide empirical evidence to show the noise injection and the gradient clipping involved in differentially private Stochastic Gradient Descent impact the overall accuracy, and especially that of underrepresented classes. The privacy framework we consider is extremely different, as the diverse impact is not a direct consequence of the privacy preservation framework design, but of the architectural adjustments made to speed up cryptographic secure inference . Such adjustments impact the approximation capabilities of the model resulting in disparate accuracy across different groups.

Finally, model compression has also been shown to induce fairness issues. Empirical observations reported that quantization, network compression, and knowledge distillation could amplify the unfairness in different learning tasks (Lukasik et al., 2022; Hooker et al., 2020a;b; Joseph et al., 2020; Blakeney et al., 2021; Ahn et al., 2022). Inspired by these works, we show how ReLU linearization techniques may adversely affect the fairness of the resulting predictors.

3. Settings

The paper considers a dataset S_T consisting of N individual data points $(\boldsymbol{x}_i, a_i, y_i)$, with $i \in [N]$ drawn i.i.d. from an unknown distribution Π. Therein, $\boldsymbol{x}_i \in \mathcal{X}$ is a feature vector, $a_i \in \mathcal{A} = [M]$, for some finite M, is a demographic group attribute, and $y_i \in \mathcal{Y}$ is a C-class label ($\mathcal{Y} = [C]$). The goal is to learn a classifier $f_{\boldsymbol{\theta}} : \mathcal{X} \to \mathcal{Y}$, where $\boldsymbol{\theta}$ is a vector of real-valued parameters. The classifier f is often defined through a *soft classifier* $h_{\boldsymbol{\theta}} : \mathcal{X} \to \mathbb{R}^C$, that, for a

given input $\boldsymbol{x}$, produces scores for each label in $\mathcal{Y}$, defining $f_{\boldsymbol{\theta}}(\boldsymbol{x}) = \operatorname{argmax}_j h_{\boldsymbol{\theta}}(\boldsymbol{x}; j)$ where $h_{\boldsymbol{\theta}}(\boldsymbol{x}; j)$ denotes the j-th component of $h_{\boldsymbol{\theta}}(\boldsymbol{x})$. The model quality is measured in terms of a loss function $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+$ and the training minimizes the empirical risk function $J(_{\boldsymbol{\theta}}; S_T)$:

$$\boldsymbol{\theta}^* = \operatorname*{argmin}_{\boldsymbol{\theta}} J(\boldsymbol{\theta}; S_T) = \frac{1}{|S_T|} \sum_{(\boldsymbol{x},a,y)\in S_T} \ell\left(h_{\boldsymbol{\theta}}(\boldsymbol{x}), y\right).$$

The paper studies the disparate impacts of classifiers $f_{\boldsymbol{\theta}}$ when it is subject to a ReLU partial reduction. The fairness notion employed is that of *accuracy parity* (Barocas et al., 2023), which holds when the classifier's misclassification rate is conditionally independent of the protected group. That is, for any $\bar{a} \in \mathcal{A}$

$$\Pr\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}) \neq y | a = \bar{a}\right) = \Pr\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}) \neq y\right).$$

In other words, this property advocates for equal errors of the classifier on different subgroups of inputs. Empirically, it is measured by comparing the accuracy rates over an evaluation set S_E comprised of data points drawn from the same distribution Π as the training set.

4. Why Network Linearization May Increase Unfairness?

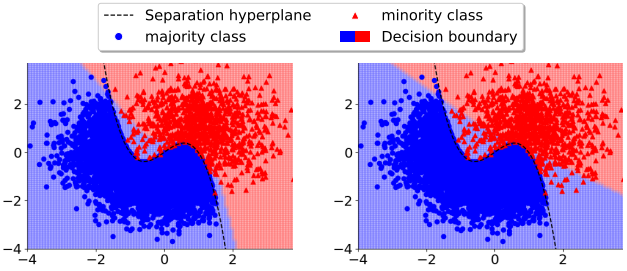
This section introduces theoretical insights to elucidate the causes behind the observed disparity effects resulting from the linearization of ReLU reduction. We start by recalling some fundamental results that will help us develop our analysis in the rest of the paper. The first results (Liu & Liang, 2021) defines a connection between the number of piecewise functions and the approximation error of convex functions. Throughout the section, $\mathcal{F}$ denotes the set of strict convex univariate functions.

Proposition 4.1. *Consider a function $f \in \mathcal{F}$ and let $f_n^\star$ be its optimal approximation through a piecewise linear function with n segments. Then, the approximation error $\Delta(f_n^\star)$ is bounded by the number of its segments, and it decreases at a rate of $O(\frac{1}{n^2})$.*

The second result (Arora et al., 2018) upper-bounds the number of pieacewise linear functions that can be represented by a ReLU network.

Proposition 4.2. *Given a ReLU $\mathbb{R} \to \mathbb{R}$ DNN with k hidden layers, and layer widths $\omega_1, \ldots, \omega_k$, the number of attainable linear pieces is at most $2^{k-1} \cdot (\omega_1 + 1) \cdot \omega_2 \cdot \cdots \cdot \omega_k$.*

This bound links the network's capability to approximate complex decision boundaries with the count of ReLU functions it contains (see Appendix B.1 for an illustrative example). Together with the previous result on the approximation error linked to a specific number of pieces (Proposition 4.1), it sets the stage for our subsequent analysis. We will illustrate that linearizing a number of ReLU functions not only leads to a potential decrease in accuracy when approximating nonlinear boundaries but it may also result in varying degrees of accuracy impact across distinct groups.



(a) Original ReLU model. (b) 50% ReLU linearization.

Figure 2. Decision boundary estimated by a ReLU model (left) and a linearized model (right). The ground truth decision boundary (black dotted line) separates majority (blue) from minority (red) samples. The linearized model shows an inferior approximation of the decision boundary, and therefore a lower prediction accuracy. Moreover, its decision boundary is deeper into to the minority class resulting in accuracy disparities.

We start with an illustrative experiment, detailed in Figure 2. It analyzes a dataset requiring a non-linear decision boundary for class separation (black dotted line), representing a protected group in fairness terms, with majority (blue) and minority (red) classes. We compare two DNN models with identical structures but different activation functions: a ReLU model and a modified model where half of the ReLU activations are replaced with linear activations, as shown in Figure 2a and Figure 2b, respectively. Notice that, after training both models from scratch, the ReLU model achieves better boundary approximation and accuracy (99.3% globally, 100.0% for the majority class, and 93.0% for the minority class), while the modified model shows comparable global accuracy (99.0%) but much lower fairness, indicated by reduced accuracy for the minority class (88.0%) compared to the original ReLU model.

4.1. Decision Boundary Approximation and Fairness

These insights indicate that there exists classification instances where the quantity of ReLU functions within a DNN enhances the approximation of decision boundaries, thereby improving prediction accuracy. Although empirical results often apply to much more complex settings than those which can be analyzed formally, they align with the forthcoming theoretical findings. In the following, we will use $\boldsymbol{\theta}$ to denote the parameters of the original ReLU network, which is assumed to have R ReLU functions, and $\tilde{\boldsymbol{\theta}}^{\boldsymbol{r}}$ those associated to the linearized model, which retains r ReLU functions. We omit the superscript when it is sufficient to denote a network retaining $1 \leq r < R$ ReLU functions.

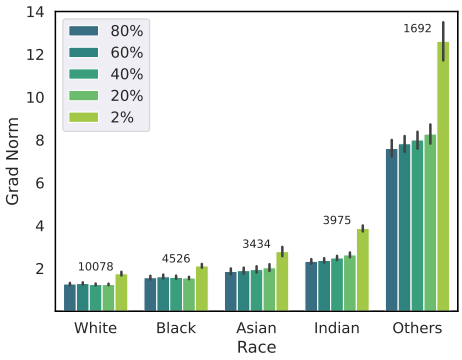


Figure 3. Grad Norms for classification on UTKFace with race labels using ResNet18 and SNL-based (Cho et al., 2022) ReLU linearization across various ReLU budgets. Subgroup sizes are reported on top of each bar group.

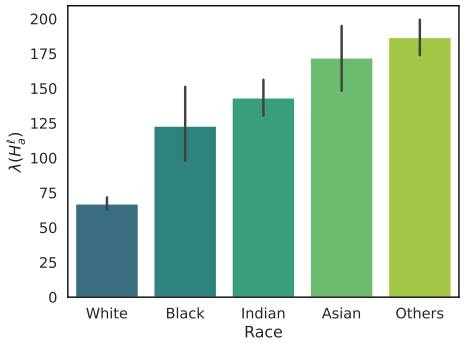


Figure 4. ResNet-18: Highest eigenvalues of Hessians for UTKFace with race labels obtained from the base model.

Proposition 4.3. *Let $f \in \mathcal{F}$ be the optimal decision boundary for a classification task, and assume that $\boldsymbol{\theta}$ implements $f_n^\star$, i.e., the minimal error approximation. For a sample set $S^a = \{(\boldsymbol{x}, \bar{a}, y) \sim \Pi | \bar{a} = a\}$ with data points drawn from Π containing exclusively members of group $a \in \mathcal{A}$, it holds*

$$R(a) = J(\tilde{\boldsymbol{\theta}}; S^a) - J(\boldsymbol{\theta}; S^a) \geq 0,$$

for any $a \in \mathcal{A}$, where all the models are assumed to be trained to achieve their best possible performance.

The above follows directly from the optimality of $\boldsymbol{\theta}$. In other words, for this class of boundaries, any degree of ReLU linearization results in models with higher loss, within the class of decision boundaries considered.

The next result is adapted from Tran et al. (2021) which exploits a similar approximation in the context of differentially private stochastic gradient descent. We upper bound the drop in loss for a given group $a \in \mathcal{A}$ due to ReLU linearization by two key interpretable components: the *gradient norm* of the samples in group a, and the *maximum eigenvalues of the Hessian* of the loss function associated with such a group.

Theorem 4.4. *For a sample set $S^a = \{(\boldsymbol{x}, \bar{a}, y) \sim \Pi | \bar{a} = a\}$ with data points drawn from Π containing exclusively members of group $a \in \mathcal{A}$, the difference between the risk functions of some protected group a of a model $\boldsymbol{\theta}$ trained on a ReLU network and one $\tilde{\boldsymbol{\theta}}$ trained on a linearized ReLU network, is bounded by:*

$$R(a) \leq \|\boldsymbol{g}_a\| \times \|\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta}\| + \frac{1}{2}\lambda\left(\boldsymbol{H}_a\right) \times \|\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta}\|^2 + O\left(\|\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta}\|^3\right), \quad (1)$$

where $\boldsymbol{g}_a = \nabla J(\boldsymbol{\theta}; S_T^a)$ describes the gradient norm of samples in group a, $\boldsymbol{H}_a^\ell = \nabla^2 J(\boldsymbol{\theta}; S_T^a)$ is the Hessian of the loss function associated with group a, and $\lambda(\Sigma)$ denotes the maximum eigenvalue of matrix Σ.

The proof is based on the Taylor expansion of the loss function around the parameters $\boldsymbol{\theta}$ and is relegated to Appendix A.2. In the above, and throughout the paper, the gradients are understood to be over the model's parameters.

There are two major takeaways from Proposition 4.2. For given parameter settings $\boldsymbol{\theta}$ and $\tilde{\boldsymbol{\theta}}$, the first component of the bound is influenced by the gradient norm of the loss function for each group $a \in \mathcal{A}$. Informally, the gradient provide information on how close the model is to local optima; larger gradients suggest sub-optimality, while smaller ones indicate proximity to these optima. Typically, in converged models, underrepresented groups tend to have larger gradient norms, implying a greater distance from local optima (cf. Appendix A.1). This aspect is highlighted in Figure 3, where each shade represents a linearized model variant with varying counts of retained ReLU functions. Here, it's evident that gradient norms are higher for underrepresented groups, decreasing as more ReLU functions are retained. Next, the bound's second term is governed by the Hessian of the loss function for each group $a \in \mathcal{A}$. The Hessian reflects the local curvature, or sharpness, of the loss function. Generally, smaller Hessian eigenvalues suggest (with certain caveats) a flatter curve for the group loss, potentially indicating better generalizability to unseen samples.

Proposition 4.5. *Consider a binary classifier $f_{\boldsymbol{\theta}}$ trained using binary cross entropy loss. For any group $a \in \mathcal{A}$ the maximum eigenvalue of the group Hessian $\lambda\left(\boldsymbol{H}_a\right)$ can be upper bounded by*

$$\lambda\left(\boldsymbol{H}_a^\ell\right) \leq Z_1 + Z_2, \quad (2)$$

where,

$$Z_1 = \frac{1}{|S^a|} \sum_{(\boldsymbol{x},a,y) \in S^a} \underbrace{h_{\boldsymbol{\theta}}(\boldsymbol{x})\left(1 - h_{\boldsymbol{\theta}}(\boldsymbol{x})\right)}_{\text{Proximity to decision boundary}} \times \left\|\nabla h_{\boldsymbol{\theta}}(\boldsymbol{x})\right\|^2,$$

$$Z_2 = \underbrace{\left|f_{\boldsymbol{\theta}}(\boldsymbol{x}) - y\right|}_{\text{Error}} \times \lambda\left(\nabla^2 h_{\boldsymbol{\theta}}(\boldsymbol{x})\right).$$

The proof (see Appendix A.2) relies on derivations of the Hessian associated with the model loss function and Weyl inequality with the notion of proximity to decision boundary is derived by (Cohen et al., 2019). As a consequence, groups with small Hessians eigenvalues (those generally distant from the decision boundary and highly accurate) tend to be less sensitive to the effects of the ReLU reduction rate. Conversely, groups with large Hessians eigenvalues tend to be affected by the ReLU reduction rate to a greater extent, typically resulting in larger excessive losses.

Figure 4 illustrates the theoretical insights presented in Proposition 4.5: crucially, the value of $\lambda\left(\boldsymbol{H}_a^\ell\right)$ corresponding to the underrepresented groups are consistently higher than those corresponding to highly represented groups.

We note that the bound in Theorem 4.4 depends also on the distance $\|\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta}\|$ between the ReLU reduced model and the original model. Generally, it is not trivial to establish a formal relationship between such distance and the rate of ReLU linearization; indeed such a formalization depends inherently on the model and the ReLU distribution induced by the linearization scheme adopted. However, our empirical analysis provides a strong sign for such trend to occur, as it will be further elaborated in Section 5.2, Figure 8.

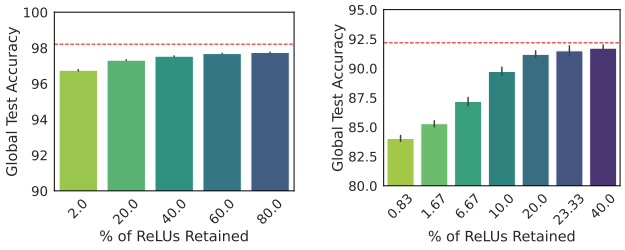
5. Empirical Results on the Fairness Analysis

We next present the key findings from our empirical analysis on how ReLU linearization affects the fairness of linearized models. These findings build upon and extend the theoretical insights from earlier sections to scenarios involving high-dimensional and non-convex decision boundaries. In summary, we show that **(1)** model linearization produces disparate degradation of performance across groups and that such disparity is enhanced with the larger amounts of linearization; **(2)** the relative magnitude of gradient norms for a group serves as a reliable indicator of the group's accuracy decline resulting from ReLU linearization; **(3)** groups located farther from (or closer to) the decision boundary are less (more) likely to experience accuracy reduction at varying levels of ReLU linearization[1].

Datasets and Models. We adopt three datasets:

- **UTKFace** (Zhang et al., 2017). Containing over 20,000 face images with annotations for age, gender, and race is widely used for computer vision tasks. Our experiments use **age** and **race** as protected groups.
- **SVHN** (Digits) (Netzer et al., 2011). Containing $60,000$ 32×32 RGB images of digits of house number plates.
- **CIFAR-10** (Krizhevsky & Hinton, 2009). Containing $60,000$ 32×32 RGB images of 10 classes (airplanes, cars, birds, cats, deer, dogs, frogs, horses, ships, and trucks).

[1]Code: https://github.com/SaswatD27/ICML_Linearization_Disparate_Impact



(a) SNL linearization (b) DR linearization

Figure 5. Global test accuracy on UTKFace with age labels using ResNet18 for SNL and DR. The red horizontal line represents the original (no ReLU linearization) global test accuracy.

To assess the effect of network linearization on groups with various representations, we evaluate several models and architectures on unbalanced datasets. While UTKFace and SVHN are naturally unbalanced, we unbalanced CIFAR-10 by selecting 90% of the samples for each of 5 classes, and retaining 10% of the samples for each of the remaining classes. All reported metrics are average over 10 random seeds. The UTKFace dataset includes images with a large variation in age, ranging from 0 to 116 years old. As customary in demography, the ages are categorized into four groups: 2-12 (young children), 13-17 (adolescents), 18-64 (working-age individuals), and over 64 (senior citizens) for the experiments in this paper and these groups contain, respectively, 10.14%, 3.63%, 77.56%, and 8.67% of the total samples in the dataset. Data points corresponding to ages 0 and 1 are considered outliers and thus omitted for this task.Race classification on UTKFace utilizes the dataset's provided labels: white, black, Asian, native Indian, and others which correspond to 42.51%, 19.09%, 14.49%, 16.77%, and 7.14% of the total samples in the dataset, respectively. Finally, digit recognition studies are conducted on the naturally imbalanced SVHN dataset. This dataset contains 10 digits from 0 to 9 which have 6.75%, 18.92%, 14.45%, 11.60%, 10.18%, 9.39%, 7.82%, 7.64%, 6.89%, and 6.36% of the total samples in the dataset, respectively

In this analysis two popular architectures are considered: ResNet18 and ResNet34 (He et al., 2015). These architectures are the main ones used in the linearization frameworks of (Jha et al., 2021; Cho et al., 2022).

Linearization Methods. The paper considers two ReLU linearization techniques, representing the current state of the art: Selective Network Linearization (SNL) (Cho et al., 2022) and DeepReDuce (DR) (Jha et al., 2021). Both methods aim to linearize ReLUs in neural networks, but their approaches differ: SNL selectively linearizes specific ReLUs, whereas DR provides for a much coarser linearization strategy. In particular, DR does not allow to control the

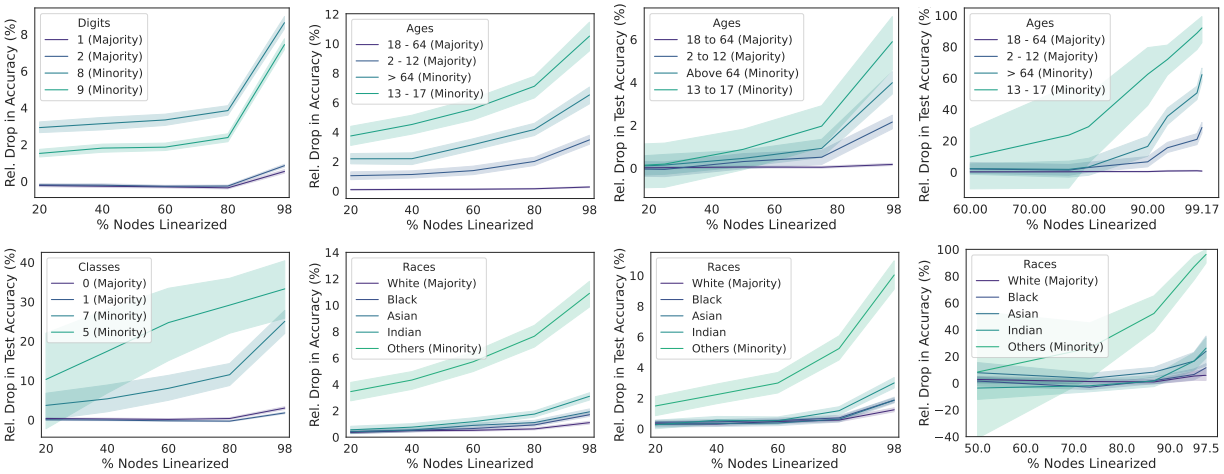


Figure 6. SNL: Relative test accuracy drop for different datasets , models, and ReLU linearization methods. **1st col:** ResNet18 trained on SVHN (top) and CIFAR-10 (bottom) with SNL; **2nd col:** ResNet18 trained on UTKFace age (top) and race (bottom) labels with SNL; **3rd col:** ResNet34 trained on UTKFace age (top) and race (bottom) labels with SNL; **4th col:** ResNet18 trained on UTKFace age (top) and race (bottom) labels with DR; The results show that performance on minority groups is affected disparately by ReLU linearization across choices of datasets, model architectures, and linearization methods.

"distribution" of the ReLU functions across the architecture, but only to linearize blocks of contiguous nodes. This fundamental difference is expected to result in distinct patterns of disparity observed with each method. As a matter of fact, DR is not inherently data-driven, and its performance is not only affected by the number of residual ReLU functions, but also by the block that the practitioner decides to linearize. In turn, this results in different linearized model configurations as reported in Figure 5b[2].

In the following, the term "ReLU budget" is used to refer to the number of ReLUs retained in a ReLU-linearized model across both methods. However, note that specifying a desired ReLU budget is feasible only with SNL. In contrast, with DR, the number of retained ReLUs depends on the specific layers linearized. Additionally, while ReLU linearization may lead to a loss in utility, notice that the models and ReLU budgets used for the analysis yield a minor decrease in global test accuracies (as shown in Figure 5) indicating that these configurations represent practical, deployable ReLU-linearized models.

5.1. Impact on Accuracy

In the primary empirical findings, the study highlights the relative decreases in test accuracy for various groups under different ReLU budgets, as shown in Figure 6. For datasets with more classes like CIFAR-10 and SVHN, the analysis focuses on the two most and two least represented classes in the test sets to maintain clarity and prevent plot overcrowding. The *relative drop in test accuracy* serves as a measure of accuracy loss due to ReLU linearization. The larger this value, the greater the performance decline. Let $\bar{J}(\boldsymbol{\theta}; S) = {}^1/_{|S|} \sum_{(\boldsymbol{x},y)\in S} \mathbb{1}\left[f_{\boldsymbol{\theta}}(\boldsymbol{x}) = y\right]$, measure the classifier $\boldsymbol{\theta}$ accuracy over set S, then the relative drop in test accuracy for group a is measured as:

$$\frac{\bar{J}(\boldsymbol{\theta}; S^a) - \bar{J}(\tilde{\boldsymbol{\theta}}_r; S^a)}{\bar{J}(\boldsymbol{\theta}; S^a)} \times 100.$$

The results reveal a consistent trend: majority groups generally show resilience to ReLU linearization while minority groups experience increasingly significant accuracy losses with more linearization. This pattern holds true across various architectures and datasets. For example, using SNL on ResNet-34 trained on UTKFace with race labels, white individuals experience almost no impact in accuracy drop across different ReLU linearization budgets, while the "other" race group suffers an almost 5-fold accuracy drop. Similarly, for DR on ResNet-18 trained on the same dataset, the accuracy for white individuals remains almost unaffected while the "other" race group suffers an almost 10-fold accuracy drop, with a near 100% accuracy drop for the lowest ReLU budget. Indeed, a similar observation holds for DR for UTKFace with age labels.

Recall that Theorem 4.4 highlighted the connection between a drop in loss (as a differentiable proxy for accuracy) experienced by a given group, under a certain ReLU linearization

[2]We also attempted to incorporate results from SENet (Kundu et al., 2023), but faced challenges. Although a validation code for the published models is available, a public implementation of the linearization algorithm is missing. Despite our efforts, we could not obtain the code or reproduce the results. Given that SENet aims to preserve global accuracy, we suspect it may encounter fairness issues similar to other frameworks considered.

level, and two key characteristics: the gradient norms associated with such groups at convergence and the distance to the decision boundary. While these results hold for a restricted classes of functions, the next two sections provide further evidence on such relationships on highly non linear models for a variety of linearization techniques and architectures. Further experiments on different architectures and wider variants of the ResNet models are reported in Figure 7.

5.2. Gradient Norms and Fairness

Firstly, we notice Theorem 4.4 highlights how both the group gradient and the group Hessian terms are multiplicatively influenced by the distance $\|\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta}\|$ between the model with reduced ReLU functions and the original model. Figure 8 (left) illustrates that this distance increases with the increase in linearized ReLU functions, suggesting that the reduction in ReLU activations amplifies the effect on the gradient norm. Next, Figure 8 (middle) presents a scatterplot that contrasts accuracy against gradient norms at the highest level of linearization examined. It reveals an inverse relationship between gradient norms and accuracy: data points for the majority group align with lower gradient norms and higher accuracy, whereas points for minority groups are linked with higher gradient norms and lower accuracy. Taken together, the increasing distance between reduced and original model parameters, alongside the varied impact on gradient norms across different groups, highlight the dynamics of gradient norms as a factor explaining the exacerbation of unfairness post-ReLU linearization.

5.3. Distance to the Decision Boundary and Fairness

Next, we look into the influence of the group Hessian on the accuracy reduction resulting from ReLU linearization. Recall that Proposition 4.5 establishes a link between the group Hessian and the distance to the decision boundary: notably, a smaller distance (or more proximity) to the decision boundary correlates with higher bounds for the associated group Hessian. This relationship renders the distance to the decision boundary an insightful, interpretable metric. Figure 8 (right) elucidates this concept. It illustrates the relationship between each test sample's distance to the decision boundary and its accuracy. The data points are color-coded according to group membership, revealing a strong correlation between the distance to the boundary and accuracy levels. Specifically, groups identified as majorities tend to have a larger distance to the boundary (and high accuracy), whereas minority groups exhibit a smaller distance.

This observation, in conjunction with what observed in Figure 8 (left), underscores the key relation between distance to the decision boundary plays and fairness issues following ReLU linearization. Understanding of this interplay is crucial to design the proposed mitigation technique, reviewed in the next section.

6. Mitigation and Impact on Fairness

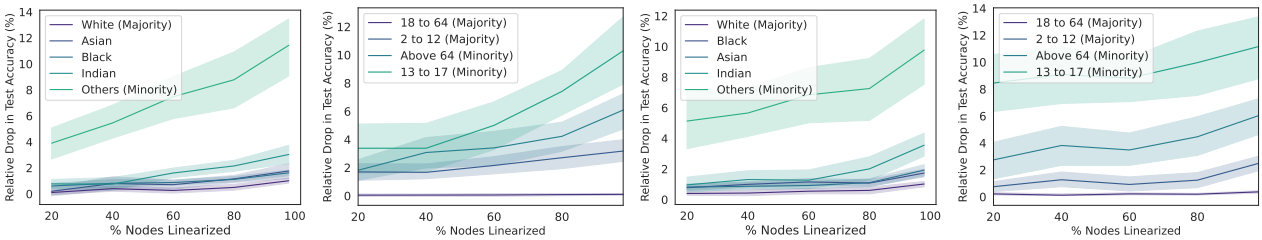
So far, we have presented an in-depth analysis of the disparate effects that linearization techniques produce on the fairness of unbalanced datasets, showing how reduced ReLU budgets negatively affect the accuracy over underreprested groups. We will now present a solution to extend ReLU linearization to contexts where fairness matters along with Private Inference. Recall from Proposition 4.5 the relationship between a group's Hessian eigenvalues, their proximity to the decision boundary, and the corresponding errors made by the model for that group. Since, as shown in the previous section, the distance to the decision boundary of a sample directly relates to their accuracy drop across various ReLU linearizaiton rates, our strategy leverages this intuition and introduces "fairness regularizers" to equalize the distances to the decision boundary of various groups. These regularizers, capture the differences in losses between groups and population, (and by proxy the distance to the boundary). The method leverages Lagrangian duality principles (Fioretto et al., 2020) for implementation and it works off-the-shelf with the Knowledge-Distillation (KD) based finetuning step, offering a seamless integration into existing algorithms.

Algorithm 1 reports the proposed fairness-aware finetuning method, replacing the standard finetuning step of SNL and DR. Let us introduce the main notation. Therein, $\mathcal{L}^{\text{KD}} : \mathbb{R}^C \times \mathbb{R}^C \times \mathbb{R}^d \to \mathbb{R}$ is the loss function associated with the KD finetuning, which, in addition to a batch of data B, takes as input two classifiers $h^r_{\tilde{\boldsymbol{\theta}}}$ and $h_{\boldsymbol{\theta}}$, and returns a scalar; $\boldsymbol{\lambda_e}$ is the vector of Lagrangian multipliers, each associated to a fairness *constraint violation*, expressed as the distance between the loss computed over a group and that computed over the whole dataset. These multipliers are updated at each epoch e; μ is the multiplier associated with the Lagrangian update step; and α is the learning rate. Furthermore, we call $\mathcal{L}^{\text{KD}}_a$ the loss function, when evaluated over the samples of a specific group $a \in \mathcal{A}$. Crucially, the proposed algorithm allows the student model, i.e., the linearized one, to be trained with knowledge flowing from the original model, while at the same time being constrained against excessive degradation of performance over underrepresented groups.

Let $\bar{J}(\tilde{\boldsymbol{\theta}}^r; S) = {}^1\!/\!_{|S|} \sum_{(\boldsymbol{x},y)\in S} \mathbb{1}\left[f_{\tilde{\boldsymbol{\theta}}^r}(\boldsymbol{x}) = y\right]$, measure the classifier θ accuracy over set S, then the relative drop in test accuracy for group a is measured as:

$$\frac{\bar{J}(\tilde{\boldsymbol{\theta}}^r; S^a) - \bar{J}(\tilde{\boldsymbol{\theta}}^r_{mit}; S^a)}{\bar{J}(\tilde{\boldsymbol{\theta}}^r; S^a)} \times 100,$$

where, with a slight abuse of notation, $\tilde{\boldsymbol{\theta}}^r$ represents the model parameter before the mitigation, and, $\tilde{\boldsymbol{\theta}}^r_{mit}$ the parameter of the corresponding linearized model finetuned with our mitigation algorithm.



(a) SNL on UTKFace (race) for VGG-19. (b) SNL on UTKFace (age) for VGG-19. (c) SNL on UTKFace (race) for WideResNet. (d) SNL on UTKFace (age) for WideResNet.

Figure 7. Results for VGG-19 and WideResNet models on UTKFace with SNL. We observe that the performance is consistent with the results obtained for the ResNet18 and ResNet34 models: more linearization produces higher accuracy loss for underrepresented groups, while the accuracy remains almost unchanged in the majority group. We observe that the phenomenon of disparity in the accuracy for underrepresented groups is evident on WideResNet-22-8, a wider variant of ResNet.

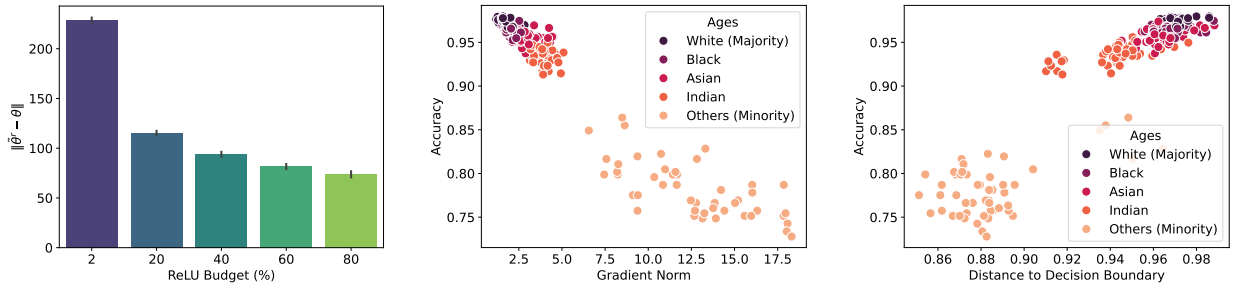


Figure 8. SNL on UTKFace with race labels and ResNet-18: **left:** $\|\tilde{\boldsymbol{\theta}}^r - \boldsymbol{\theta}\|$ vs. ReLU budget; **middle:** group accuracy vs. gradient norm for 2% ReLUs retained; **right:** group accuracy vs. distance to decision boundary for 2% ReLUs retained.

Algorithm 1 Fairness Mitigation for ReLU Linearization

Require: $h_{\tilde{\boldsymbol{\theta}}^r}$, $h_{\boldsymbol{\theta}}$, Multiplier $\mu > 0$, step size η,
loss function $\mathcal{L}^{\text{KD}}$, group-wise loss function $\mathcal{L}_a^{\text{KD}}$,
dataset S_T, set of groups $\mathcal{A}$
$\boldsymbol{\lambda}_1 \leftarrow (0)_{a\in\mathcal{A}}$ {array of zeros with size $|\mathcal{A}|$}
for epoch $e = 1, 2, \cdots$ **do**
  for all mini-batch $B \subseteq S_T$ **do**
    $\rho_{\tilde{\boldsymbol{\theta}}^r,\boldsymbol{\theta},B} \leftarrow |\mathcal{L}^{\text{KD}}(f_{\tilde{\boldsymbol{\theta}}^r}, f_{\boldsymbol{\theta}}, B) - \mathcal{L}_a^{\text{KD}}(f_{\tilde{\boldsymbol{\theta}}^r}, f_{\boldsymbol{\theta}}, B)|$
    $\tilde{\boldsymbol{\theta}}^r \leftarrow \tilde{\boldsymbol{\theta}}^r - \alpha\nabla_{\tilde{\boldsymbol{\theta}}^r}\left(\mathcal{L}(f_{\tilde{\boldsymbol{\theta}}^r}, f_{\boldsymbol{\theta}}, B) + \boldsymbol{\lambda}_e^\top \rho_{\tilde{\boldsymbol{\theta}}^r,\boldsymbol{\theta},B}\right)$
  end for
  $\boldsymbol{\lambda}_{e+1} \leftarrow \boldsymbol{\lambda}_e + \mu|\mathcal{L}^{\text{KD}}(f_{\tilde{\boldsymbol{\theta}}^r}, f_{\boldsymbol{\theta}}, S_T) - \mathcal{L}_a^{\text{KD}}(f_{\tilde{\boldsymbol{\theta}}^r}, f_{\boldsymbol{\theta}}, S_T)|$
end for

Figure 9 illustrates the effect of the mitigation strategy on the UTKFace dataset for both SNL (left) and DR (right) methods, with results segmented by age (top) and race (bottom) labels. Negative values indicate accuracy improvements for a group compared to the baseline accuracy of the original model at the same level of ReLU reduction. There are two key aspects: First, the mitigation consistently enhances accuracy across all groups and linearization levels tested, with minimal negative effects on majority classes. Second, notice how the minority groups improve their relative performance, a trend that is especially evident for DR at higher levels of ReLU linearization. Furthermore, there is an observable reduction in gradient norms and an augmentation in the distance to the decision boundary for minority groups under the mitigation approach (see Appendix D.1). These results illustrate the potential for effective mitigations to significantly improve the performance for minority groups without detriment to majority groups, thus substantially reducing the unfairness produced by ReLU linearization.

Finally, an ablation study on the impact of the parameter μ in the mitigation algorithm is presented in Appendix D.2.

7. Discussion

Figure 9 demonstrates the effectiveness of the mitigation method, highlighting the importance of choosing the right approach for practitioners aiming to enforce fairness predictably. A data-driven ReLU linearization method like SNL, which carefully considers the impact of linearization on model utility and fairness, stands out as preferable over simpler methods such as DR. In DR, ReLUs may be linearized arbitrarily across any layer without regard for potential impacts on the model performance.

This is evident in Figure 6, where the smallest groups under DR exhibit significantly poorer performance compared to

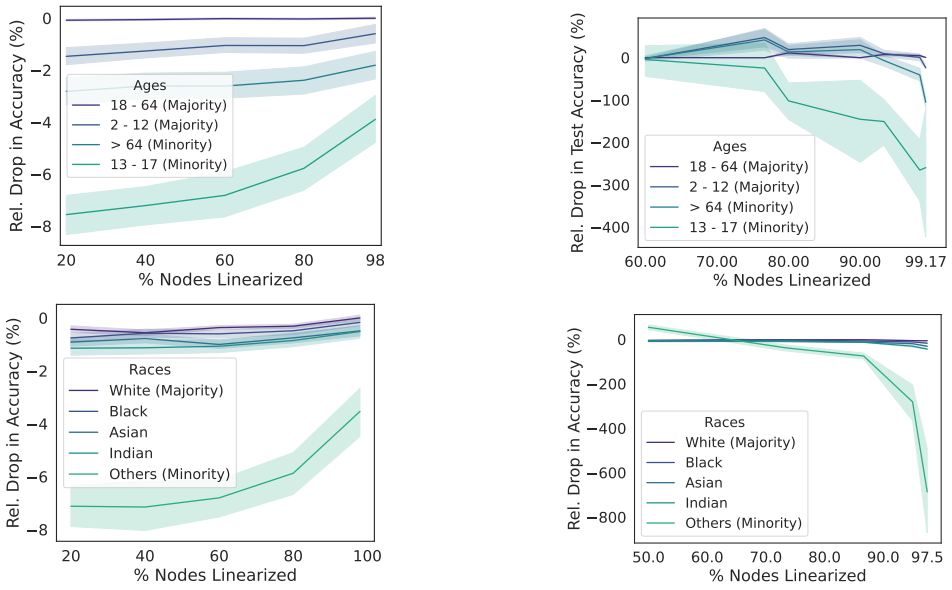


Figure 9. Mitigation: Relative test accuracy drop for different linearization methods. **left:** ResNet18 trained on UTKFace age (top) and race (bottom) labels with SNL; **right:** ResNet18 trained on UTKFace age (top) and race (bottom) labels with DR.

others and when SNL is applied. The unpredictable utility and fairness outcomes with DR make it a less suitable option for achieving fair models predictably. In contrast, SNL offers consistent and predictable outcomes, enabling practitioners to more easily apply fair ReLU linearization and select suitable parameters for effective mitigation.

Limitations. Our theoretical framework proposes a bound for the residual loss of a DNN model and its linearized counterpart, elucidating their relationship with group norms, group Hessians, and the norm of the weight distance between the two models' parameters. While our analysis introduces proxies to articulate the connection between group size, linearized ReLU proportion, and accuracy drop, examining the actual relationship between these quantities remains an open question. Addressing this gap presents an avenue for future research, with an opportunity to formalize fundamental limits of disparate impact mitigation strategies through non-trivial lower bounds on loss differences. We note that the proposed mitigation strategy necessitates access to protected group information. Although this information is frequently available, the requirement may restrict the applicability of the proposed mitigation in contexts where such information may be missing. To address this limitation, future research could explore the integration of our mitigation approach with tailored loss functions to promote fairness without relying on demographic information (cf. Lahoti et al. (2020)) or with a context of distributionally robust objectives (cf. Hashimoto et al. (2018)).

8. Conclusion

This study observed that while ReLU linearization strategies effectively reduce computational costs and inference times, they inadvertently exacerbate accuracy disparities across different subgroups, particularly affecting underrepresented ones. This phenomenon was found consistent across various datasets and architectures. The empirical analysis was grounded in theoretical insights, which highlighted two key factors responsible for the observed unfairness. We showed that as ReLU functions are approximated, the gradient for underrepresented groups tend to increase, suggesting that the model becomes more sensitive to input variations for these groups, leading to a higher misclassification rates. This effect is exacerbated for underrepresented groups due to their typically smaller distance from the decision boundary in the space that these models operate within. This distance to the boundary, when combined with altered gradient norms due to ReLU linearization, results in a disproportionate impact on the fairness of the model's outcomes.

Motivated by such observations, the paper proposed a mitigation solution which acts on regularizing the finetuning step of ReLU linearization strategies through a Lagrangian dual approach. This simple solution was found effective in balancing the computational benefit of ReLU linearization with the imperative of fairness.

Acknowledgements

This research is partially supported by NSF grants 2133169, 2232054 and NSF CAREER Award 2143706. Fioretto is also supported by an Amazon Research Award and a Google Research Scholar Award. Its views and conclusions are those of the authors only.

Impact Statement

Our contributions shed light on the Matthew effect within accuracy reduction caused by ReLU linearization, offering a mathematical intuition of the observed disparities, demonstrating the widespread presence of these effects across algorithms and models, and proposing a viable mitigation strategy. The proposed strategy not only addresses fairness concerns but also manages to preserve overall accuracy to a commendable extent.

The evidence presented, particularly the stark accuracy disparities illustrated in Figure 1, calls for a reevaluation of model optimization strategies, especially in the use of datasets reflecting real-world diversity. The findings emphasize the necessity for approaches that consider the differential impact of linearization on various subgroups, advocating for a more inclusive and fair machine learning ecosystem.

References

Ahn, J., Lee, H., Kim, J., and Oh, A. Why knowledge distillation amplifies gender bias and how to mitigate from the perspective of distilbert. *Proceedings of the 4th Workshop on Gender Bias in Natural Language Processing (GeBNLP)*, 2022.

Arora, R., Basu, A., Mianjy, P., and Mukherjee, A. Understanding deep neural networks with rectified linear units. In *6th International Conference on Learning Representations, ICLR*, 2018.

Bagdasaryan, E., Poursaeed, O., and Shmatikov, V. Differential privacy has disparate impact on model accuracy. In Wallach, H. M., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E. B., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 15453–15462, 2019.

Barocas, S., Hardt, M., and Narayanan, A. *Fairness and Machine Learning: Limitations and Opportunities*. MIT Press, 2023.

Blakeney, C., Huish, N., Yan, Y., and Zong, Z. Simon says: Evaluating and mitigating bias in pruned neural networks with knowledge distillation. *ArXiv*, abs/2106.07849, 2021.

Cho, M., Joshi, A., Reagen, B., Garg, S., and Hegde, C. Selective network linearization for efficient private inference. In *International Conference on Machine Learning, ICML*, 2022.

Cohen, J., Rosenfeld, E., and Kolter, J. Z. Certified adversarial robustness via randomized smoothing. In *International Conference on Machine Learning*, pp. 1310–1320. PMLR, 2019.

Cybenko, G. Approximation by superpositions of a sigmoidal function. *Math. Control. Signals Syst.*, 1989.

Farrand, T., Mireshghallah, F., Singh, S., and Trask, A. Neither private nor fair: Impact of data imbalance on utility and fairness in differential privacy. In Zhang, B., Popa, R. A., Zaharia, M., Gu, G., and Ji, S. (eds.), *PPMLP'20: Proceedings of the 2020 Workshop on Privacy-Preserving Machine Learning in Practice, Virtual Event, USA, November, 2020*, pp. 15–19. ACM, 2020. doi: 10.1145/3411501.3419419.

Fioretto, F., Van Hentenryck, P., Mak, T. W. K., Tran, C., Baldo, F., and Lombardi, M. Lagrangian duality for constrained deep learning. In *European Conference on Machine Learning*, volume 12461 of *Lecture Notes in Computer Science*, pp. 118–135. Springer, 2020. doi: 10.1007/978-3-030-67670-4_8.

Ghodsi, Z., Veldanda, A. K., Reagen, B., and Garg, S. Cryptonas: Private inference on a relu budget. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

Hanin, B. and Rolnick, D. Deep relu networks have surprisingly few activation patterns. In Wallach, H. M., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E. B., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 359–368, 2019.

Hashimoto, T. B., Srivastava, M., Namkoong, H., and Liang, P. Fairness without demographics in repeated loss minimization. In Dy, J. G. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1934–1943. PMLR, 2018.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2015.

Hooker, S., Courville, A. C., Clark, G., Dauphin, Y., and Frome, A. What do compressed deep neural networks forget. *arXiv: Learning*, 2020a.

Hooker, S., Moorosi, N., Clark, G., Bengio, S., and Denton, E. L. Characterising bias in compressed models. *ArXiv*, abs/2010.03058, 2020b.

Hornik, K. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4(2):251–257, 1991. doi: 10.1016/0893-6080(91)90009-T.

Jha, N. K., Ghodsi, Z., Garg, S., and Reagen, B. Deepreduce: Relu reduction for fast private inference. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pp. 4839–4849. PMLR, 2021.

Joseph, V., Siddiqui, S. A., Bhaskara, A., Gopalakrishnan, G., Muralidharan, S., Garland, M., Ahmed, S., and Dengel, A. R. Going beyond classification accuracy metrics in model compression. 2020.

Krizhevsky, A. and Hinton, G. Learning multiple layers of features from tiny images. Technical Report 0, University of Toronto, Toronto, Ontario, 2009.

Kundu, S., Lu, S., Zhang, Y., Liu, J. T., and Beerel, P. A. Learning to linearize deep neural networks for secure and efficient private inference. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023.

Lahoti, P., Beutel, A., Chen, J., Lee, K., Prost, F., Thain, N., Wang, X., and Chi, E. H. Fairness without demographics through adversarially reweighted learning. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

Lin, H. and Jegelka, S. Resnet with one-neuron hidden layers is a universal approximator. In Bengio, S., Wallach, H. M., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pp. 6172–6181, 2018.

Liu, B. and Liang, Y. Optimal function approximation with relu neural networks. *Neurocomputing*, 435:216–227, 2021. doi: 10.1016/J.NEUCOM.2021.01.007.

Lou, Q., Shen, Y., Jin, H., and Jiang, L. Safenet: A secure, accurate and fast neural network inference. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021.

Lu, Z., Pu, H., Wang, F., Hu, Z., and Wang, L. The expressive power of neural networks: A view from the width. In Guyon, I., von Luxburg, U., Bengio, S., Wallach, H. M., Fergus, R., Vishwanathan, S. V. N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pp. 6231–6239, 2017.

Lukasik, M., Bhojanapalli, S., Menon, A. K., and Kumar, S. Teacher's pet: understanding and mitigating biases in distillation. *Transactions on Machine Learning Research*, 2022. ISSN 2835-8856.

Mishra, P., Lehmkuhl, R., Srinivasan, A., Zheng, W., and Popa, R. A. Delphi: A cryptographic inference service for neural networks. In Capkun, S. and Roesner, F. (eds.), *29th USENIX Security Symposium, USENIX Security 2020, August 12-14, 2020*, pp. 2505–2522. USENIX Association, 2020.

Montúfar, G., Pascanu, R., Cho, K., and Bengio, Y. On the number of linear regions of deep neural networks. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. D., and Weinberger, K. Q. (eds.), *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pp. 2924–2932, 2014.

Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., and Ng, A. Y. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011*, 2011.

Pan, X. and Srikumar, V. Expressiveness of rectifier networks. In Balcan, M. and Weinberger, K. Q. (eds.), *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, volume 48 of *JMLR Workshop and Conference Proceedings*, pp. 2427–2435. JMLR.org, 2016.

Shen, Z., Yang, H., and Zhang, S. Optimal approximation rate of relu networks in terms of width and depth. *Journal de Mathématiques Pures et Appliquées*, 157:101–135, 2022. ISSN 0021-7824. doi: https://doi.org/10.1016/j.matpur.2021.07.009.

Tran, C., Dinh, M., and Fioretto, F. Differentially private empirical risk minimization under the fairness lens. In *Advances in Neural Information Processing Systems*, volume 34, pp. 27555–27565. Curran Associates, Inc., 2021.

Yarotsky, D. Error bounds for approximations with deep relu networks. *Neural Networks*, 94:103–114, 2017. doi: 10.1016/J.NEUNET.2017.07.002.

Zhang, Z., Song, Y., and Qi, H. Age progression/regression by conditional adversarial autoencoder. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017.

A. Additional Theoretical Results

A.1. Underrepresented groups have larger gradients.

Let us assume $a, b \in \mathcal{A}$ such that with $|S^a| \geq |S^b|$. Then $\left\|\boldsymbol{g}_a^\ell\right\| \leq \left\|\boldsymbol{g}_b^\ell\right\|$.

Proof. Let us assume convergence of the model with parameters $\boldsymbol{\phi}$ to a (local) minimum. Then, it holds that:

$$\begin{aligned}\nabla\mathcal{L}(\boldsymbol{\phi}; S) &= \sum_{a\in\mathcal{A}} \frac{|S^a|}{|S|} \nabla J\left(\boldsymbol{\phi}; S^a\right) \\ &= \frac{|S^a|}{|D|}\boldsymbol{g}_a^\ell + \frac{|S^b|}{|D|}\boldsymbol{g}_b^\ell = \mathbf{0}\end{aligned}$$

Thus, $\boldsymbol{g}_a^\ell = -\frac{|S^b|}{|S^a|}\boldsymbol{g}_b$. Hence $\left\|\boldsymbol{g}_a^\ell\right\| = \frac{|S^b|}{|S^a|}\left\|\boldsymbol{g}_b^\ell\right\| \leq \left\|\boldsymbol{g}_b^\ell\right\|$, because $|S^a| \geq |S^b|$.

□

A.2. An upper-bound for the residual loss.

Theorem 1. The excessive loss of a group $a \in \mathcal{A}$ is upper bounded by

$$R(a) \leq \left\|\boldsymbol{g}_a^\ell\right\| \times \|\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta}\| + \frac{1}{2}\lambda\left(\boldsymbol{H}_a^\ell\right) \times \|\tilde{\boldsymbol{\theta}} - \hat{\boldsymbol{\theta}}\|^2 + O\left(\|\tilde{\boldsymbol{\theta}} - \hat{\boldsymbol{\theta}}\|^3\right),$$

where $\boldsymbol{g}_a^\ell = \nabla J\left(\boldsymbol{\theta}; S^a\right)$ is the vector of gradients associated with the loss function ℓ evaluated at $\boldsymbol{\theta}$ and computed using group data S^a, $\boldsymbol{H}_a^\ell = \nabla^2 J\left(\boldsymbol{\theta}; S^a\right)$ is the Hessian matrix of the loss function ℓ, at the optimal parameters vector $\boldsymbol{\theta}$, computed using the group data S^a (henceforth simply referred to as group hessian), and $\lambda(\Sigma)$ is the maximum eigenvalue of a matrix Σ.

Proof. Using a second order Taylor expansion around $\boldsymbol{\theta}$, the excessive loss $R(a)$ for a group $a \in \mathcal{A}$ can be stated as:

$$\begin{aligned}R(a) &= J\left(\tilde{\boldsymbol{\theta}}; S^a\right) - J\left(\boldsymbol{\theta}; S^a\right) \\ &= \left[J\left(\boldsymbol{\theta}; S^a\right) + (\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta})^\top \nabla J\left(\boldsymbol{\theta}; S^a\right) + \frac{1}{2}(\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta})^\top \boldsymbol{H}_a^\ell (\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta}) + O\left(\|\boldsymbol{\theta} - \tilde{\boldsymbol{\theta}}\|^3\right)\right] - J\left(\boldsymbol{\theta}; S^a\right) \\ &= (\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta})^\top \boldsymbol{g}_a^\ell + \frac{1}{2}(\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta})^\top \boldsymbol{H}_a^\ell (\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta}) + O\left(\|\boldsymbol{\theta} - \tilde{\boldsymbol{\theta}}\|^3\right)\end{aligned}$$

The above, follows from the loss $\ell(\cdot)$ being at least twice differentiable, by assumption. By Cauchy-Schwarz inequality, it follows that

$$(\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta})^\top \boldsymbol{g}_a^\ell \leq \|\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta}\| \times \left\|\boldsymbol{g}_a^\ell\right\|.$$

In addition, due to the property of Rayleigh quotient we have:

$$\frac{1}{2}(\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta})^\top \boldsymbol{H}_a^\ell (\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta}) \leq \frac{1}{2}\lambda\left(\boldsymbol{H}_a^\ell\right) \times \|\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta}\|^2$$

A.3. An upper-bound for the maximum eigenvalue of the group Hessian

Let $f_{\boldsymbol{\theta}}$ be a binary classifier trained using a binary cross entropy loss. For any group $a \in \mathcal{A}$, the maximum eigenvalue of the group Hessian $\lambda\left(\boldsymbol{H}_a^\ell\right)$ can be upper bounded by:

$$\lambda\left(\boldsymbol{H}_a^\ell\right) \leq \frac{1}{|S^a|} \sum_{(\boldsymbol{x},y)\in S^a} \underbrace{\left(h_{\boldsymbol{\theta}}(\boldsymbol{x})\right)\left(1 - h_{\boldsymbol{\theta}}(\boldsymbol{x})\right)}_{\text{Proximity to decision boundary}} \times \|\nabla h_{\boldsymbol{\theta}}(\boldsymbol{x})\|^2 + \underbrace{\left|f_{\boldsymbol{\theta}}(\boldsymbol{x}) - y\right|}_{\text{Accuracy}} \times \lambda\left(\nabla^2 h_{\boldsymbol{\theta}}(\boldsymbol{x})\right).$$

Proof. First notice that an upper bound for the Hessian loss computed on a group $a \in \mathcal{A}$ can be derived as:

$$\lambda\left(\boldsymbol{H}_a^\ell\right) = \lambda\left(\frac{1}{|S^a|}\sum_{(\boldsymbol{x},y)\in S^a}\boldsymbol{H}_x^\ell\right) \leq \frac{1}{|S^a|}\sum_{(\boldsymbol{x},y)\in S^a}\lambda\left(\boldsymbol{H}_x^\ell\right) \tag{3}$$

where $\boldsymbol{H}_{\boldsymbol{x}}^\ell$ represents the Hessian loss associated with a sample $\boldsymbol{x} \in S^a$ from group a. The above follows Weily's inequality which states that for any two symmetric matrices A and B, $\lambda(A+B) \leq \lambda(A) + \lambda(B)$. Next, we will derive an upper bound on the Hessian loss associated to a sample $\boldsymbol{x}$. First, based on the chain rule a closed form expression for the Hessian loss associated to a sample $\boldsymbol{x}$ can be written as follows:

$$\boldsymbol{H}_{\boldsymbol{x}}^\ell = \nabla^2\ell\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y\right)\left[\nabla f_{\boldsymbol{\theta}}(\boldsymbol{x})\left(\nabla f_{\boldsymbol{\theta}}(\boldsymbol{x})\right)^\top\right] + \nabla\ell\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y\right)\nabla^2 f_{\boldsymbol{\theta}}(\boldsymbol{x}).$$

The next follows from that

$$\begin{aligned}\nabla\ell\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y\right) &= \left(f_{\boldsymbol{\theta}}(\boldsymbol{x}) - y\right),\\ \nabla^2\ell\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y\right) &= f_{\boldsymbol{\theta}}(\boldsymbol{x})\left(1 - f_{\boldsymbol{\theta}}(\boldsymbol{x})\right).\end{aligned}$$

Applying the Weily inequality again on the R.H.S. of Equation 12, we obtain:

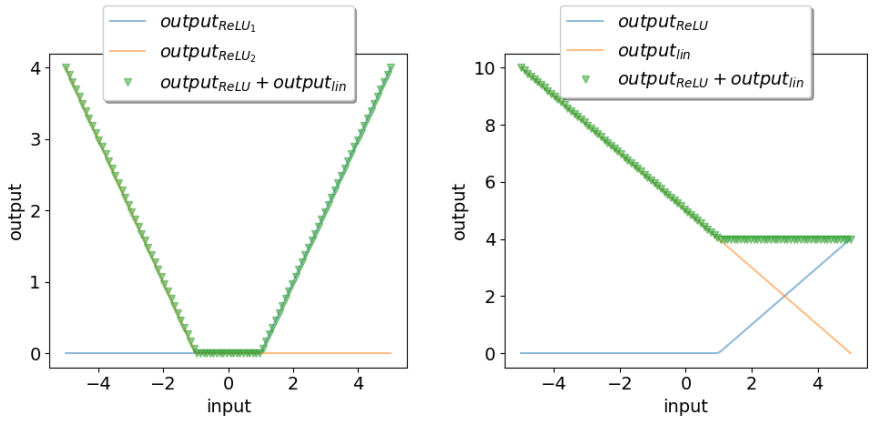
$$\begin{aligned}\lambda\left(\boldsymbol{H}_{\boldsymbol{x}}^\ell\right) &\leq f_{\boldsymbol{\theta}}(\boldsymbol{x})\left(1 - f_{\boldsymbol{\theta}}(\boldsymbol{x})\right) \times \left\|\nabla f_{\boldsymbol{\theta}}(\boldsymbol{x})\right\|^2 + \lambda\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}) - y\right) \times \nabla^2 f_{\boldsymbol{\theta}}(\boldsymbol{x})\\ &\leq f_{\boldsymbol{\theta}}(\boldsymbol{x})\left(1 - f_{\boldsymbol{\theta}}(\boldsymbol{x})\right) \times \left\|\nabla f_{\boldsymbol{\theta}}(\boldsymbol{x})\right\|^2 + \left|f_{\boldsymbol{\theta}}(\boldsymbol{x}) - y\right| \lambda\left(\nabla^2 f_{\boldsymbol{\theta}}(\boldsymbol{x})\right)\end{aligned} \tag{4}$$

The statement of Proposition 4.5 is obtained combining Equation (3) with Equation (4).

□

B. Examples

B.1. On linearized models and the upper-bound in Proposition 4.2: an empirical standpoint.



(a) Model with 2 ReLU nodes.

(b) Model with 1 ReLU node and a linearized one.

Figure 10. The ReLU network in Figure 10a can attain at most 3 linear pieces, while the linearized network in Figure 10b can attain at most 2 linear pieces.

In general, linearized models may be far away from reaching the bound in Proposition 4.2. It is easy to see how this bound cannot be reached in many cases when one or more ReLU activation are replaced with linear activation functions. For

instance let us consider a ReLU network with a single hidden layer and width $\omega_1 = 2$. According to Proposition 4.2, this network can attain at most 3 linear pieces. Indeed, let us consider bounded inputs $\boldsymbol{x} \in [-b, b]$ such that $b \in \mathbb{R}^+$. By properly optimizing weights and biases, the two ReLU nodes can create the orange and blue linear pieces in Figure 10a, that, when linearly combined in the output of the model (cf. green line in Figure 10a), can attain at most 3 linear pieces. On the contrary, if one of the two ReLU nodes is replaced with a linear node, the linear combination of the two nodes which is forwarded to the output, by definition, does not create any new breakpoint, only maintaining the two linear pieces dictated by the presence of the ReLU node. Therefore, the number of linear pieces is reduced to 2, as shown in Figure 10b.

C. Additional Empirical Results

C.1. Settings

Each of the results in this paper was produced using an A100 GPUs with 80 GB of GPU memory, up to 100 GB of RAM, and up to 10 Intel(R) Xeon(R) E5-2630 v3 CPUs each clocked at 2.40GHz.

For SNL, we train each base model for 160 epochs, and then perform the SNL finetuning step after ReLU linearization. These values are present in the official implementation of this algorithm provided by the authors of (Cho et al., 2022), and we use this implementation off-the-shelf.

For DeepReDuce, we use the official implementation provided by the authors of (Jha et al., 2021) off-the-shelf which trains each model for 200 epochs.

C.2. Accuracy drop analysis

For SVHN and CIFAR-10, the classes corresponding to these labels are evident. The class correspondences for UTKFace with age and race labels are as follows.

- **UTKFace with age labels:** 2-12, 13-17, 18-64, and over 64 years of age correspond to 1, 2, 3, and 4, respectively.
- **UTKFace with race labels:** White, black, Asian, Indian, and other races correspond to 0, 1, 2, 3, and 4, respectively.

In this section, we present more plots for relative accuracy drops in Figure 11 to show the disparate impact of linearizing ReLUs. Observe again how minority groups suffer degradation in accuracy that is higher than for majority groups as the ReLU budget decreases/more ReLUs are linearized.

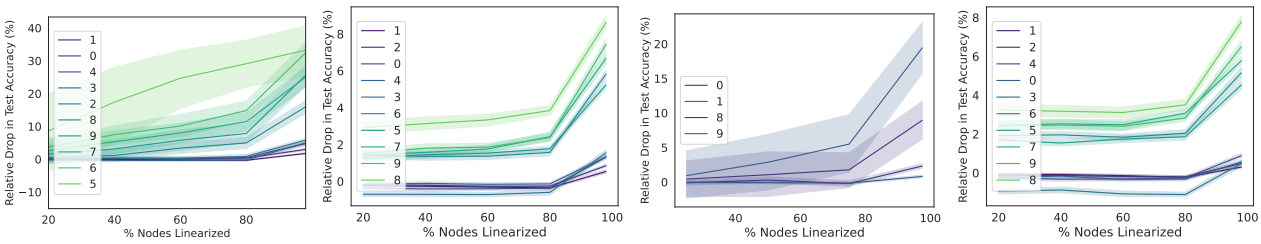


Figure 11. SNL: For ResNet18 on CIFAR-10 (left) and SVHN (center left), and for ResNet34 on CIFAR-10 (center right) and SVHN (right)

D. Mitigation

This subsection provides further evidence for the efficacy of the mitigation method discussed in Section 6. The labels in the legend refer to group indices.

D.1. Mitigation: Gradient Norms and Distance to the Decision Boundary

The plots in Figure 13 and Figure 14 illustrate the differences in the values of gradient norms and (normalized) distances to the decision boundary when linearizing ReLUs with and without reduction. As mentioned in Section 6, a reduction in the gradient norms and an increase in the distance to the decision boundary can be observed when using the mitigation method.

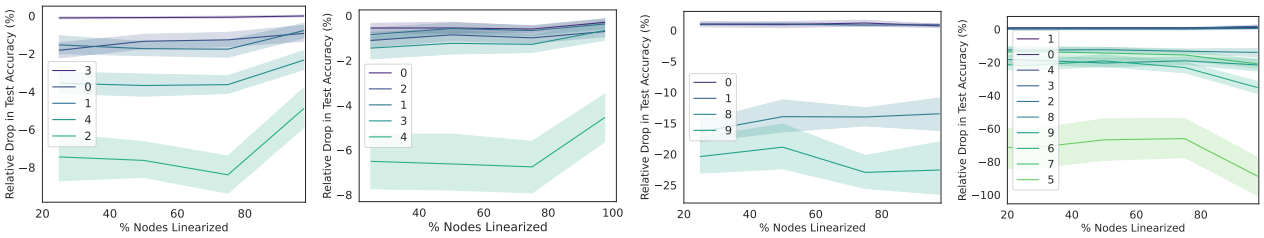


Figure 12. SNL with Mitigation: For ResNet34 trained on UTKFace with age (left) and race (center left) labels and on CIFAR-10 (center right) and ResNet18 trained on CIFAR-10 (right).

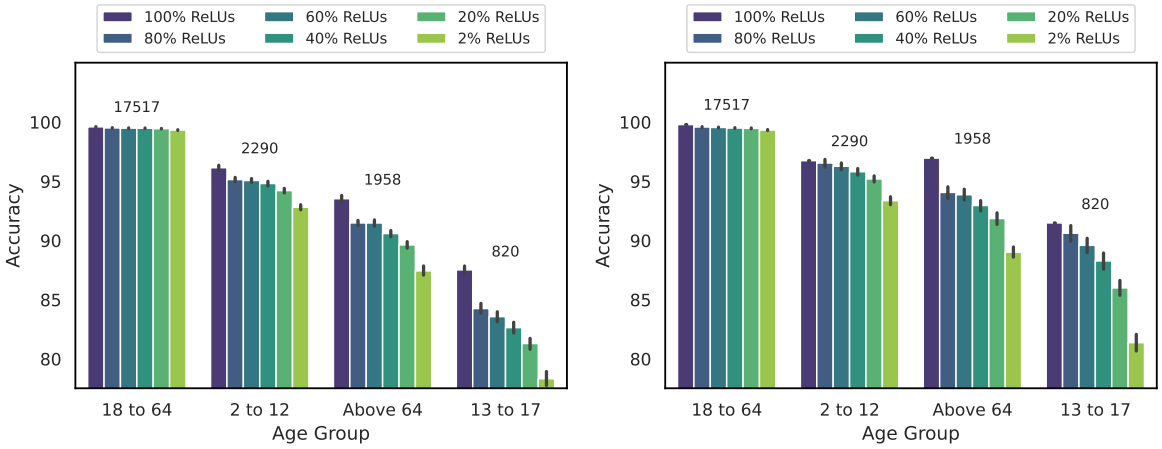


Figure 13. SNL: (Test) Accuracies for ResNet-18 trained on UTKFace with age labels; **left:** without mitigation; **right:** with mitigation.

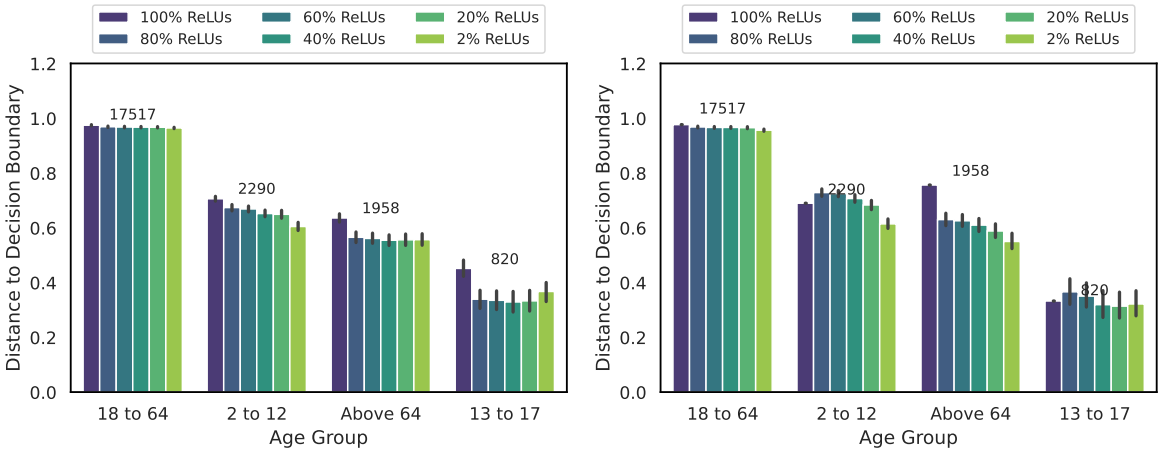
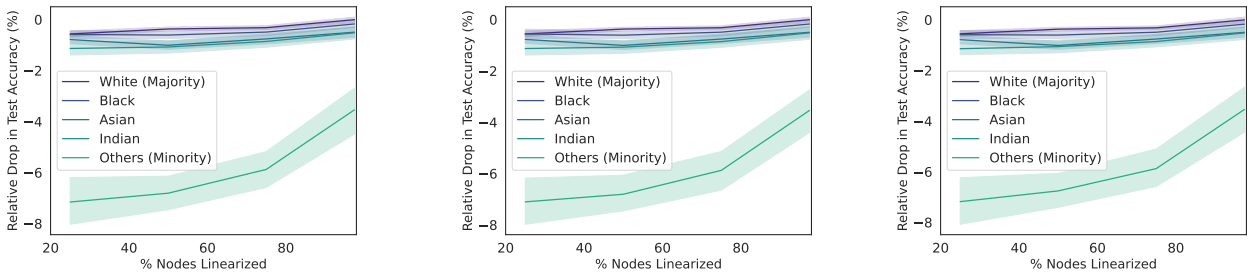


Figure 14. SNL: (Normalized) Distances to Decision Boundary for ResNet-18 trained on UTKFace with age labels; **left:** without mitigation; **right:** with mitigation.

D.2. Mitigation: an ablation study of the parameter μ



(a) SNL on UTKFace race ResNet18 mitigation with $\mu = 0.0005$

(b) SNL on UTKFace race ResNet18 mitigation with $\mu = 0.005$

(c) SNL on UTKFace race ResNet18 mitigation with $\mu = 0.01$

Figure 15. We report the plots for the ablation study of the parameter μ for the SNL on UTKFace (race). We consider $\mu \in \{0.0005, 0.005, 0.01\}$. In general we observe that these values of μ do not result in significant changes for the performance of the mitigation strategy. For the task at end, we could then say that there exists a neighborhood of values for μ where our mitigation consistently benefits the considered linearization techniques.