Fair Information Bottleneck; Mitigating Unfairness in kNN Classifiers

Alexander Yeung
ayeung@umass.edu

Shreyans Babel
sbabel@umass.edu

Harold Thidemann
hthidemann@umass.edu

Abstract

Machine learning models, particularly those used in societal applications, are prone to biased predictions when trained on datasets containing sensitive demographic information. We investigate pre-processing techniques to mitigate these biases specifically for k-Nearest Neighbor (kNN) classifiers. We are working towards a novel embedding generation method that reduces the amount of sensitive demographic information in the dataset, while aiming to retain prediction accuracy. By modifying the information bottleneck objective to emphasize compression on sensitive attributes, our method aims to train a variational autoencoder (VAE) such that its latent space vector, when used as embeddings minimize bias from kNN predictions. We evaluate our approach on the German Credit and Adult Income datasets, comparing the results against kNN classifiers trained on embeddings produced with other baseline pre-processing techniques, including fairness-agnostic autoencoders and existing fairness-based embeddings.

1. Introduction

Machine learning models are deployed in a variety of societal applications, however, training sets for these models often include demographic information, which risks creating models which inadvertently make biased or discriminatory predictions. Therefore, an important area of research are pre-processing techniques which mitigate the effects of sensitive training information on model prediction, thus creating more fair and equitable machine learning models. In this paper, we will study the effect of these techniques on the predictions of k-Nearest Neighbor (kNN) classifiers, which are a particularly exciting to research in this context due to their high levels of interpretability and low computational cost.

To study the extent to which kNN classifiers make biased predictions based on training data, we use the following training pipeline: First, we select widely used datasets that contain sensitive demographic information. Second, we pre-process the data in various ways. As a baseline, we compare our implementation to training a kNN on the raw data without any pre-processing, on the embeddings of an autoencoder [4], and on the fairness embedding technique originally developed by Zemel et al. [14]. We will then compare each of these baselines to our embedding technique, which modifies the loss function the variational autoencoder (VAE) method developed by Alemi et al. [1]. Lastly, we evaluate the extent to which our embedding approach affected the downstream accuracy and discrimination of kNN classifiers trained on the embeddings. Our goal for these embeddings is to minimize discrimination as much as possible, whilst minimally degrading the resulting model's accuracy. To provide some visualization of the effects that our model has on these embeddings, Section 4 contains visualizations from our use of the t-SNE clustering algorithm [13] to visualize our embeddings, and provide intuition as to how our model is impacting downstream performance of kNNs.

Let us formalize the problem we are seeking to solve as follows: given a dataset D, how do we generate embeddings D_{emb} such that a kNN classifier trained on D_{emb} retains comparable accuracy to one trained on D, whilst removing bias in its predictions? It is worthwhile to note that, decreasing bias in predictions often involves removing sensitive demographic information in the dataset, resulting in a decrease in accuracy. This is a tradeoff that we will have to balance throughout the development of our method and its analysis.

To measure bias in the predictions of the kNN classifier, we use the discrimination metric employed by Zemel et al. [14], that describes the absolute difference the probability of outcomes between two groups in a binary classification task:

$$Discrimination = \left| P(\hat{Y} = 1 \mid A = 0) - P(\hat{Y} = 1 \mid A = 1) \right|$$

Where $\hat{Y} \in \{0, 1\}$ represents the predicted label, where $\hat{Y} = 1$ denotes a favorable outcome, and let $A \in \{0, 1\}$ represent the sensitive attribute, where $A = 0$ and $A = 1$ correspond to membership to one of two distinct groups (e.g., different demographic groups).

We find that the method that we develop performs comparably to the above defined task as the fairness embedding

technique developed by Zemel et al. [14], but may not perform as well as more modern techniques developed by Kenfack et al. [7], Louizos et al. [9], Madras et al. [10], Peltonen et al. [11] which use adversarial approaches or noise injection into the latent space of variational autoencoders amongst other techniques that our implementation does not use. Thus the main contributions of this paper are:

- The development of a new fairness-aware variational autoencoder structure which generates embeddings which greatly lower the discrimination of kNN classifiers whilst modestly impacting their accuracy
- Comparison of the performance of kNNs when trained on a dataset without any pre-processing, embeddings from an autoencoder without any notion of fairness, embeddings from the technique developed by Zemel et al. [14], and our newly developed technique
- Visualizations that provide some intuition as to why the embedding techniques help prevent kNNs from making discriminatory predictions

2. Related Work

Typically, we select the class A for the discrimination metric introduced in Section 1 based on demographic information for which making biased predictions would be particularly undesirable, for instance sex, race, or age. The datasets that we have selected are the German Credit Dataset [5] and the Adult Income Dataset [2], both of which are used commonly throughout the fairness literature. The German Credit Dataset [5] is formulated as a binary prediction task to determine whether a person has "Good" or "Bad" credit, and contains sensitive information each person's sex. The Adult Income Dataset [2] is also a binary prediction task where the predictions labels indicate whether or not a person's income exceeds $50,000, and contains sensitive information about each person's age. As was done in Kamiran and Calders [6], Zemel et al. [14], we divide the age categories into two groups, where one group are those above the age of 25 and the other are ages 25 and below.

The challenge for our project will be generating embeddings such that the discrimination of the predictions on each dataset is minimized, but the resulting accuracy of the kNN remains largely unaffected. Previous works in this space include Zemel et al. [14] which mapped the training data to probability distributions that removed information about sensitive attributes, Kenfack et al. [7], Louizos et al. [9] which use interesting properties about the latent space of autoencoders [4] to remove sensitive information, and Peltonen et al. [11] which used supervised dimensionality reduction techniques to eliminate sensitive demographic information.

Some implementations of embedding techniques such as Alemi et al. [1], Louizos et al. [9] make use of variational autoencoders (VAEs) which were originally introduced by Kingma and Welling [8]. In contrast to the autoencoders proposed by Hinton and Salakhutdinov [4], VAEs embed inputs into probability distributions in the latent space as opposed to single points. Modifications of these distributions can prove to be useful in the fairness literature, as oftentimes these distributions can be tweaked to exclude information about sensitive demographics, as was done in Louizos et al. [9].

Our novel technique is based off the work of Alemi et al. [1], who developed a variational approximation of the information bottleneck method initially developed by Tishby and Zaslavsky [12]. Alemi et al. [1] used their approach to generate embeddings which would improve the accuracy of image classifiers and make them more robust to adversarial attacks, but for our purposes, we modify their loss function to emphasize compression on sensitive information.

3. Methods

As mentioned in Section 2, our novel fairness embedding technique is a modification of the methods developed by Alemi et al. [1] which involves training a variational autoencoder (VAE). In contrast to other fairness methods such as Louizos et al. [9], Madras et al. [10], which seek to minimize Maximum Mean Discrepancy [3] between specified sensitive attributes, our implementation will be based on the information bottleneck approach developed by Tishby and Zaslavsky [12].

The approach outlined by Alemi et al. [1] seeks to create embeddings Z, from a data set X and labels Y with model θ such that information between the embeddings and data set is preserved. Information as a metric is described as:

$$I(Z, Y; \theta) = \int dx\ dy\ p(z, y|\theta) \log \frac{p(z, y|\theta)}{p(z|\theta)p(y|\theta)}$$

To generate their embeddings, they train a VAE to minimize the loss function:

$$\mathcal{L}(\theta) = \beta I(X; Z) - I(Z; Y)$$

where the objective for the embedding is to "learn an encoding Z that is maximally expressive about Y while being maximally compressive about X, where $\beta \geq 0$ controls the tradeoff" [1]. For our purposes, we modify this objective function to also seek to maximally compress scaled information about the sensitive attributes in our dataset. Let X_{mask} be a data set with the same shape as X, but with all non-sensitive information set equal to 0, or masked out. Our alternative approach, therefore, uses the loss function:

$$\mathcal{L}(\theta) = \beta I(X; Z) + \gamma I(Z, X_{mask}; \theta) - I(Z; Y)$$

All other variables remaining equal, the parameter γ allows us to modulate the extent to which the autoencoder

seeks to remove information about sensitive attributes. In our modification of the Alemi et al. [1] methods, we will largely use the same VAE model structure, but will modify the loss function to include a larger emphasis on removing sensitive information as described above. A comparison of how the parameters β and γ impact model accuracy and discrimination is included in Section 4.

Alemi et al. [1] highlight that their exact information bottleneck loss function may be intractable to compute over certain data distributions. Therefore, they employ an approximation to their loss function when training their autoencoder:

$$\begin{aligned}\mathcal{L}(\theta) = \beta \mathbb{E}_{p(X)}\big[\mathrm{KL}(q(Z|X)\|p(Z))\big] \\ - \mathbb{E}_{p(X,Y)}\big[\mathbb{E}_{q(Z|X)}[\log p(Y|Z)]\big]\end{aligned} \tag{1}$$

- $q(Z|X)$: Variational approximation to the true posterior distribution of Z given X.
- $p(Z)$: Prior distribution over the latent representation Z, typically chosen as $\mathcal{N}(0, I)$ (a standard normal distribution).
- $p(Y|Z)$: Predictive distribution for Y given Z, parameterized by the decoder network.
- $\mathrm{KL}(\cdot\|\cdot)$: Kullback-Leibler divergence, measuring the divergence between two distributions.

The terms in the loss function serve specific purposes as described by the authors where:

$$\beta \mathbb{E}_{p(X)}\big[\mathrm{KL}(q(Z|X)\|p(Z))\big]$$

measures how much information the latent representation Z retains about the input X. By penalizing the KL divergence between $q(Z|X)$ and $p(Z)$, this term encourages Z to be independent of X while adhering to $p(Z)$. The following term:

$$-\mathbb{E}_{p(X,Y)}\big[\mathbb{E}_{q(Z|X)}[\log p(Y|Z)]\big]$$

maximizes the log-likelihood of Y given Z, ensuring that the compressed representation Z does not remove all of the information relevant for predicting Y.

Equivalently, we use a similar approximation, but modified to add the sensitive attribute component:

$$\begin{aligned}\mathcal{L}(\theta) = \beta \mathbb{E}_{p(X)}\big[\mathrm{KL}(q(Z|X)\|p(Z))\big] \\ + \gamma \mathbb{E}_{p(X_{mask})}\big[\mathrm{KL}(q(Z|X_{mask})\|p(Z))\big] \\ - \mathbb{E}_{p(X,Y)}\big[\mathbb{E}_{q(Z|X)}[\log p(Y|Z)]\big]\end{aligned} \tag{2}$$

By adding the term:

$$\gamma \mathbb{E}_{p(X_{mask})}\big[\mathrm{KL}(q(Z|X_{mask})\|p(Z))\big]$$

to our loss function, we add a penalty to the KL divergence between $q(Z|X_{mask})$ and $p(Z)$, which we hope will encourage the embedding Z to be independent of the sensitive information attributes X_{mask}.

Our approach encourages the embeddings to be maximally compressive about not only the original dataset, modulated by the parameter β, but also the specific and targeted sensitive information attributes, modulated by the parameter γ. In Section 4 we show visualizations of the embeddings generated from autoencoders trained using our loss function, juxtaposing them to the original dataset, as well as other fair embedding techniques, which provide useful insight into how our approach helps prevent biased predictions from kNNs. These modifications show how the information bottleneck loss function, originally developed by Alemi et al. [1], were modified in our implementation to bottleneck information pertaining to sensitive attributes.

4. Results

We implemented several baselines with which to compare our novel implementation to. We measured the accuracy and discrimination on all baselines, measuring how those metrics changed as a function of k in the kNN classifier. We repeat these measurements for the German Credit [5] and the Adult Income [2] datasets.

Our baselines include unmodified kNN predictions, kNN predictions on embeddings generated from an autoencoder with no notion of fairness, and kNN predictions on embeddings generated from the work of Zemel et al. [14] (an encoder with a notion of fairness). These are compared against each other and our method as described in Section 3.

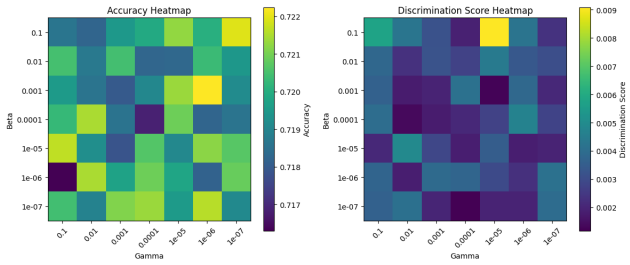

Figure 1. Accuracy and Discrimination as a result of changing the hyperparameters β and γ on Adult Income Dataset [2]

In order for our model to obtain the best results, we performed a grid search over the hyperparameters β and γ while keeping track of the associated accuracy and discrimination, this can be seen in the provided heat maps 1. We used the results of this search to set the hyperparameters as $\beta = 0.01$ and $\gamma = 0.1$.

To examine the results of our experiments, we did so in two ways over each dataset; the first being graphs showing the performance of each baseline and our model with respect to accuracy and discrimination (shown in Figures 2, 4, 3, 5), the second being clustering graphs (shown in Fig-

ures 6, 7, 8, 9). For the figures in which we report the accuracy and discrimination results of kNN classifier on each dataset, we plot the mean results of 5 different randomized trials using error bars where applicable.

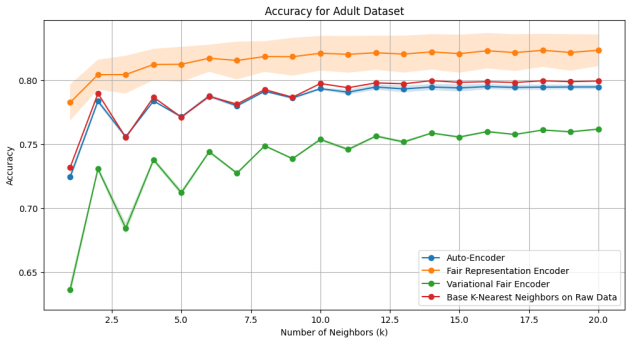

Figure 2. Accuracy vs. Number of Neighbors on the Adult Income [2] Dataset

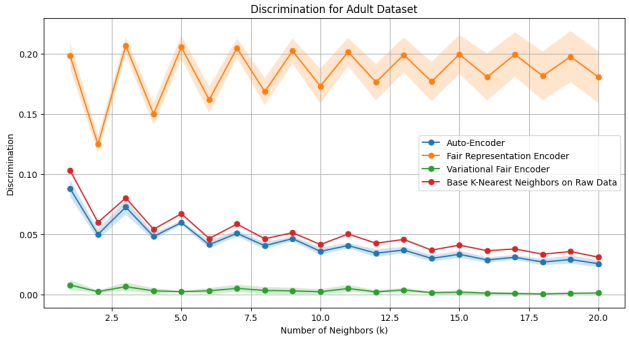

Figure 3. Discrimination vs. Number of Neighbors on the Adult Income [2] Dataset

Let us examine the results on the Adult Income [2] Dataset. We see in Figure 2 the models achieve varying degrees of accuracy with the fair representation model [14] resulting in the highest accuracy and our VAE model resulting in the lowest comparative accuracy. The fairness unaware autoencoder and kNN on the raw data achieve almost the same results. This tells us little about the models unless we also look at the discrimination scores in Figure 3. In this we see our model achieves the lowest discrimination, while the fair representation model gets the highest, again with the fairness unaware autoencoder and kNN on the raw data achieving almost the same results. We believe the results of the fair representation model are from our implementation prioritizing accuracy over discrimination from our choice of hyperparameter for this model. We see the opposite for our model as it achieves almost no discrimination while having the lowest accuracy. We are not surprised to see the fairness unaware autoencoder and base kNN having on par results as the autoencoder is embedding the data without any changes to it, ensuring it is representing the data as is.

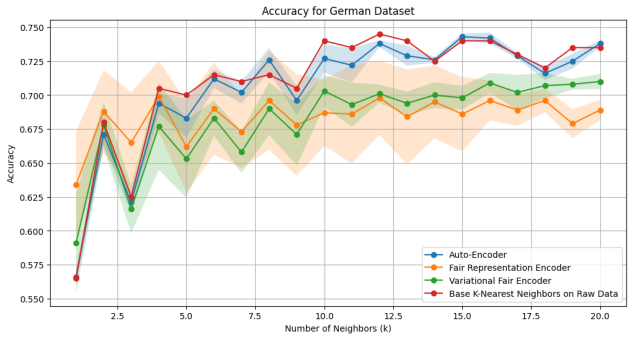

Figure 4. Accuracy vs. Number of Neighbors on the German Credit [5] Dataset

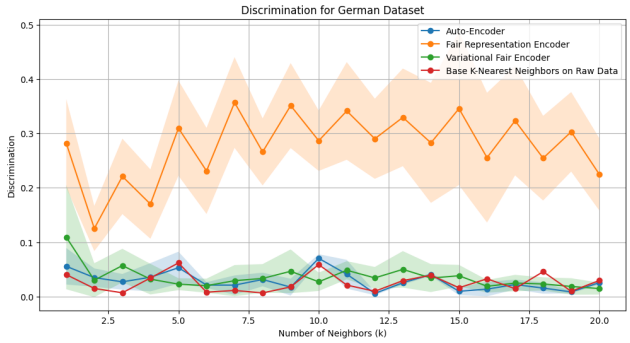

Figure 5. Discrimination vs. Number of Neighbors on the German Credit [5] Dataset

Let us examine the results on the German Credit [5] Dataset. We see in the accuracy and discrimination graphs (Figures 4 and 5 respectively), contain somewhat more sporadic results when compared against the Figures 2 and 3 for the Adult Income Dataset. When looking at the error bars we see that there is more variance in what any given model might predict in. However, we do see trends of the models generally performing better as the number of neighbors increases. Looking at the the fair representation model, our re-implementation of [14], we see discrimination is much greater than base kNN which should not be the case; our thoughts about why this happens is discussed about in section 5. As for the other models we see discrimination remains consistently low independent of the model. This points to room for improvement of our model as it obtains the same low level of discrimination without achieving the same accuracy scores.

Now let's examine the cluster graphs. For each dataset there are two graphs, one of clustering over the raw data, and one of clustering over our VAE's embedded data. For the Adult Income dataset we hvae Figures 6 and 7, and for the German Credit dataset we have Figures 8 and 9. Look-

ing at the graphs of the raw data (Figures 6, 8) we see that there is a correlation between the target classification and sensitive attribute. It is hard to say what this correlation represents but it does hint towards a correlation between the target classification and the sensitive attribute. Looking at the graphs our VAE's embedded data (Figures 7, 9) we see no clear clusters have formed, with our model making the embedding space more uniform. That is, we can see there are no clear clusters with regard to the target classification or sensitive attributes (this would be seen by groupings of similar colors in the graph, which are not there), allowing us to see the sensitive attribute with our embedding does not influence the target classification. This improves fairness as we force the sensitive attributes to be less influential, thus reducing bias.

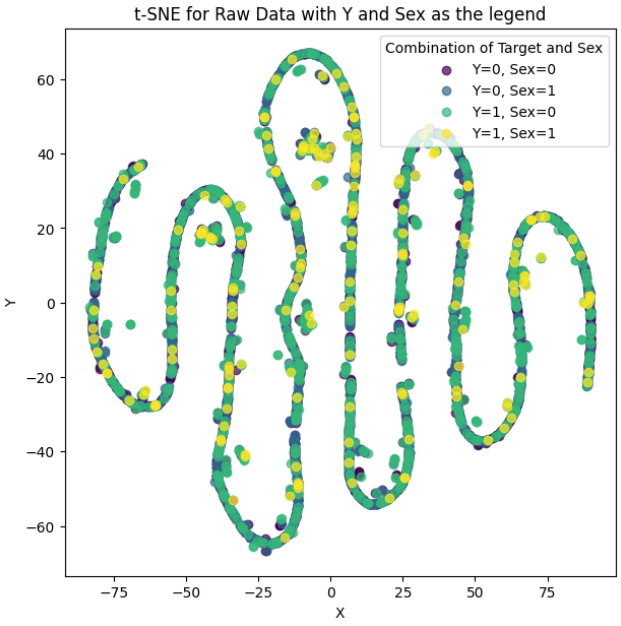

Figure 6. t-SNE clustering on the Raw Adult Income [2] Dataset

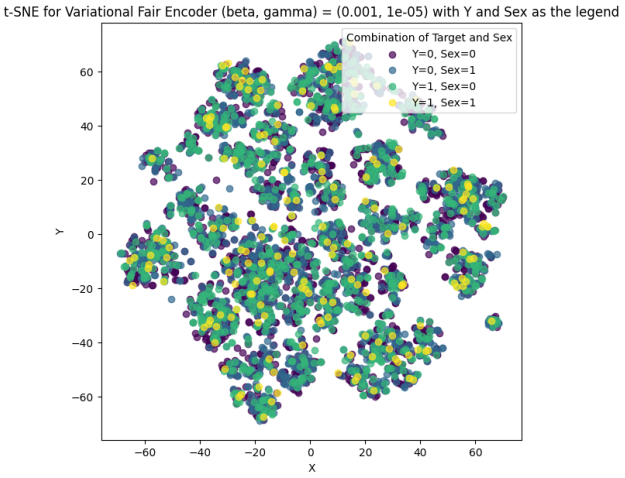

Figure 7. t-SNE clustering on the Variational Fair Encoder for Adult Income [2] Dataset

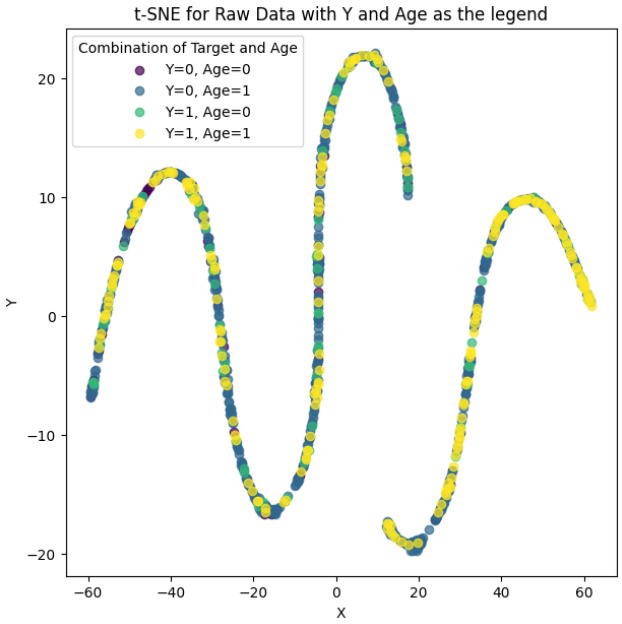

Figure 8. t-SNE clustering on the Raw German Credit [5] Dataset

5. Discussion

In this section we discuss the overall findings of our results, and some potential threats to their validity. These include the confusing results that we get in our re-implementation of the embedding technique from Zemel et al. [14], as well as the results from our hyperparameter search.

In re-implementing the technique from Zemel et al. [14], we struggled to find any sources online where they published their existing code base. Due to this, we were left to try to reimplement their technique based off the attempts of others to do so, and from their paper alone. We believed that Zemel et al. [14] was a good baseline for us to compare to, since it was one the earlier works in this space, and is cited by many other works in the literature. That being said, the results in our recreation are confusing, leading us to believe that their may be an issue in our reimplementation. For example, whilst Figures 2 and 4 show that the accuracy improves when training a kNN model on the embeddings from our reimplementation, we fail to see why in Figures 3 and 5 that these kNNs are subsequently *more* discriminatory than just training a kNN model on the raw dataset. Especially since the original paper reports that on these datasets they were effectively able to reduce discrimination by a significant amount, this points to there being some significant er-

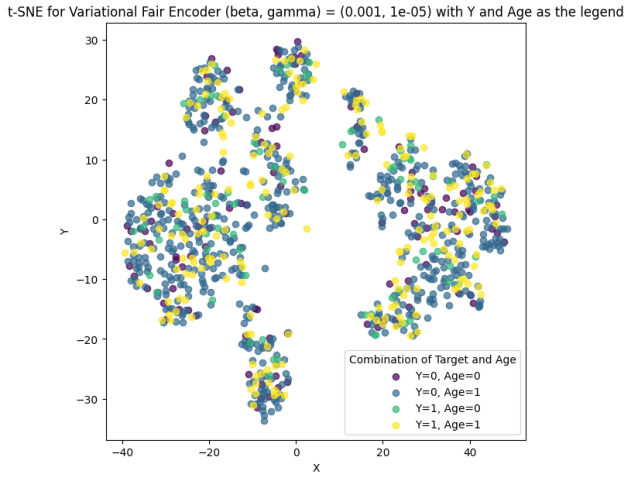

Figure 9. t-SNE clustering on the Variational Fair Encoder for German Credit [5] Dataset

ror in our code for this technique. However, in the interest of time, and after many iterations of attempting to debug our own code, we were not able to accurately recreate this technique, so it remains to be seen if this is an error on our side, or merely just a facet of training kNNs on these types of embeddings, where other models may not be so discriminatory.

We are also somewhat confused by the results from our hyperparameter search of β and γ for our VAE implementation. In theory, these hyperparameters should provide us with some degree of control over the model compression on the entire dataset, as well as the sensitive attribute. As such, since β dictates the extent to which the model is compressive about the entire dataset, we initially anticipated the accuracy would decrease as β increases, but this is not necessarily a trend that we see in the heatmap provided in Figure 1. Additionally, while we would expect that since γ dictates the extent to which the model is compressive about the sensitive attributes, we would anticipate that as γ increases, discrimination would decrease. However, this hypothesis is also not supported by Figure 1. That being said, whilst Figure 1 may reveal that our hyperparameters do not provide us with as much fine-grained control over the behavior of our model as we may like, there is sufficient evidence to show that our model does in fact play a decent role in reducing discrimination and information about sensitive attributes within the dataset. A quick glance at the embeddings from t-SNE in Section 4 shows that sensitive attributes are far more diffuse once they have been embedded using our VAE. Thus, while it appears that we cannot express great control over our model's ability to compress sensitive information, our notion of fairness does appear to work to produce less biased datasets nonetheless.

Overall, we believe that this implementation does in fact do what it initially sought out to do, which is reduce the bias in the predictions from a kNN trained on its embeddings, whilst retaining accuracy. That being said, some additional things to look into to compare model performance are additional baselines or debugging our implementation of the work of Zemel et al. [14], as well as potentially expanding the hyperparameter search space to see if greater ranges of hyperparameters could provide us with better control over model behavior, as well as increase model performance.

6. Conclusion

Our main goal for this project was to develop a novel implementation of an encoding technique which would help remove sensitive information from a dataset, preventing kNN classifiers from producing discriminatory results. To measure and compare the performance of our technique, we chose to compare against multiple baselines for the accuracy and discrimination metrics defined in Section 1. As baselines we chose to compare our approach to training a kNN on the raw German and Adult Income datasets, embeddings those aforementioned datasets using a traditional autoencoder, and embedding them using an approach discovered by Zemel et al. [14]. With our Fair Information Bottleneck based approach, we achieved results which preserved accuracy and mitigated discrimination to an extent which is comparable to what was achieved by Zemel et al. [14].

The implmentation that we developed in this work was a modification of the variational autoencoder developed by Alemi et al. [1]. In particular, we modify their loss function to add a notion of fairness. While their loss function was focused on being maximally compressive about the information in the dataset, we alter the loss function to be especially compressive about sensitive demographic information.

There are multiple future works in this space which would help provide further insight into the efficacy of our approach. First, we would like to test these results against other datasets containing sensitive demographic information, further examining how our technique works in various domain spaces. Second, we would like to compare our results to other approaches that are more modern than Zemel et al. [14]. For instance, Madras et al. [10] uses an adversarial training approach and Louizos et al. [9] uses an alternative technique to train a variational autoencoder. We would like to see if our method behaves comparably to these approaches, or if there are situations where different techniques are more well-suited to different datasets. Lastly, we would like to see if these embeddings can be useful for applications outside of training a kNN on them. For instance, we might find that these embeddings also help remove bias when used to train a multi-layer-perceptron on a similar prediction task.

References

[1] Alexander A. Alemi, Ian Fischer, Joshua V. Dillon, and Kevin Murphy. Deep variational information bottleneck. *CoRR*, abs/1612.00410, 2016. 1, 2, 3, 6

[2] Barry Becker and Ronny Kohavi. Adult. UCI Machine Learning Repository, 1996. DOI: https://doi.org/10.24432/C5XW20. 2, 3, 4, 5

[3] Arthur Gretton, Karsten Borgwardt, Malte J. Rasch, Bernhard Scholkopf, and Alexander J. Smola. A kernel method for the two-sample problem, 2008. 2

[4] G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786): 504–507, 2006. 1, 2

[5] Hans Hofmann. Statlog (German Credit Data). UCI Machine Learning Repository, 1994. DOI: https://doi.org/10.24432/C5NC77. 2, 3, 4, 5, 6

[6] Faisal Kamiran and Toon Calders. Classifying without discriminating. *2009 2nd International Conference on Computer, Control and Communication*, pages 1–6, 2009. 2

[7] Patrik Joslin Kenfack, Adín Ramírez Rivera, Adil Mehmood Khan, and Manuel Mazzara. Learning fair representations through uniformly distributed sensitive attributes. In *2023 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, pages 58–67, 2023. 2

[8] Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2022. 2

[9] Christos Louizos, Kevin Swersky, Yujia Li, Max Welling, and Richard Zemel. The variational fair autoencoder, 2017. 2, 6

[10] David Madras, Elliot Creager, Toniann Pitassi, and Richard Zemel. Learning adversarially fair and transferable representations, 2018. 2, 6

[11] Jaakko Peltonen, Wen Xu, Timo Nummenmaa, and Jyrki Nummenmaa. Fair neighbor embedding. In *ICML*, pages 27564–27584, 2023. 2

[12] Naftali Tishby and Noga Zaslavsky. Deep learning and the information bottleneck principle, 2015. 2

[13] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9 (86):2579–2605, 2008. 1

[14] Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. Learning fair representations. In *Proceedings of the 30th International Conference on Machine Learning*, pages 325–333, Atlanta, Georgia, USA, 2013. PMLR. 1, 2, 3, 4, 5, 6