Mitigating Bias in Skin Lesion Classification Models Using Variational Autoencoders

Bias Behebung in Modellen zur Hautläsionsklassifikation mithilfe von Variational Autoencoders

Jacob Schäfer

Universitätsbachelorarbeit
zur Erlangung des akademischen Grades

Bachelor of Science
(B. Sc.)

im Studiengang
IT Systems Engineering
eingereicht am 29. Juni 2023 am
Fachgebiet Algorithm Engineering der
Digital-Engineering-Fakultät
der Universität Potsdam

Gutachter	Prof. Dr. Tobias Friedrich
Betreuer	Stefan Neubert
	Jonathan Gadea Harder

Abstract

Leveraging deep learning for early detection of skin cancer could help prevent deaths. Current skin lesion classification algorithms include biases and perform worse for patients with rarer skin features. An existing bias mitigation method automatically detects rare skin features in a dataset using a Variational Autoencoder and takes them into account when training a classifier. We propose an adaptation of this method that allows having multiple classes. We show that the adaptation is effective in experiment setups similar to those in previous research. Bias with respect to age and skin tone of the patient was successfully reduced by more than 45%, with a significance of $p < 0.0005$. Further, we observe that using transfer learning diminishes the bias mitigation effects while providing decreased biases on its own. Lastly, we find that the method is not effective for a more complex multi-class skin lesion classification task. We discuss potential reasons and areas for future work.

Zusammenfassung

Die Anwendung von Deep Learning in der Früherkennung von Hautkrebs kann dazu beitragen, die Anzahl an Todesfällen zu verringern. Aktuelle Algorithmen zur Hautläsionsklassifikation funktionieren schlechter für Patient:innen mit selteneren Hautmerkmalen. Eine Methode um dem entgegenzuwirken, ist mithilfe eines Variational Autoencoders automatisch seltene Hautmerkmale zu erkennen, um diese beim Training des Algorithmus zu berücksichtigen. Wir stellen eine Anpassung dieser Methode vor, die auf Probleme mit mehreren Klassen angewendet werden kann.

Wir zeigen die Effektivität unserer Methode, indem wir ein Experiment durchführen, welches in ähnlicher Form bereits in verwandten Arbeiten durchgeführt wurde. Mit einer Signifikanz von $p < 0.0005$ konnten wir Bias in Bezug auf das Alter und den Hautton der Patient:innen um mehr als 45% reduzieren.

In einem weiteren Experiment beobachten wir, dass Transfer Learning die Effektivität der Biasreduktion verringert, dabei aber bereits für sich allein zu einer Reduktion beiträgt. Schließlich stellen wir fest, dass die Methode für komplexere Klassifizierungsprobleme mit mehreren Klassen nicht funktioniert. Wir diskutieren potenzielle Gründe und Bereiche für zukünftige Forschung.

Contents

1 Introduction

Over the last decades, skin cancer has become increasingly common. This trend is expected to continue [WHO17]. In 2020, there have been more than 300,000 incidents of melanoma skin cancer worldwide. These incidents resulted in about 60,000 deaths [WCR22].

However, mortality varies greatly depending on the stage at which the cancer is diagnosed. For instance, if melanoma is diagnosed when it has not spread to other parts of the skin, the five-year survival rate is higher than 99%. This rate is reduced to 32 % if melanoma has already spread to distant parts of the body at the time of diagnosis [Soc23]. Therefore, it is of public interest to improve methods for early detection.

In the past years, a lot of research has been conducted to automatically diagnose skin lesions based on dermoscopic skin images. The International Skin Imaging Collaboration (ISIC) has collected a large benchmark dataset of such images and organized several challenges with the goal to ”improve dermatologic diagnostic accuracy with the aid of AI” [ISI23a]. One of these challenges was to classify dermoscopic images into seven classes of different types of skin lesions. The submissions for this challenge have shown high accuracy on the unseen test data [Cod+19].

However, a significant decrease in performance is observed for images containing particular skin features that are observed less frequently. This was measured on the top 25 submissions for the mentioned ISIC challenge. For instance, pigmented melanomas are correctly classified 71% of the time while non-pigmented melanomas are correctly classified only 46% of the time. In order to be able to safely deploy such algorithms in clinical processes, these biases need to be addressed [Com+22].

This problem of having biases is not limited to the domain of skin cancer diagnostics. Consider, for instance, gender classification, where the task is to determine the gender of a person based on an image. Algorithms from Microsoft and IBM both showed better performance on subjects with lighter skin in comparison to subjects with darker skin [BG18].

The origin of such biases is often found in the data used to train the algorithm. Algorithms may reflect or even amplify the unequal distribution of training samples across different demographic groups [Cam+17].

In the past years, several methods have been developed to deal with unbalanced data and ultimately mitigate bias.

One approach is to augment the dataset with synthetic data to achieve balance with respect to specific features. This can be achieved using generative adversarial networks as proposed by Abusitta et al. [AAW20], Celis and Keswani [CK19], and Xu et al. [Xu+18]. However, this method requires that the data is labeled with respect to the attributes that should be debiased. If this is not the case, labeling the data can be time-consuming and expensive, depending on the specific attribute and the size of the dataset.

This shortcoming has been addressed by a different approach, which performs adaptive resampling based on an automatically detected feature distribution. First, a Variational Autoencoder (VAE) is used to learn a latent space that represents common features of images in the dataset. These features are then used to adaptively calculate sample probabilities of the individual images, such that images with rare features are sampled more often. Since rare features are extracted automatically, there is no need for additional labeling of the dataset [Ami+19].

This method has been applied to a binary skin lesion classification task by Das [Das21]. A small dataset of about 1000 images was used to train an algorithm to distinguish between melanoma and other benign skin lesions. The classification was integrated into the encoder of the VAE and was based on a simple five-layer Convolutional Neural Network (CNN). Initially, a plain version of this CNN was trained without any debiasing techniques applied. Using the same hyperparameters, the classifier was then trained adaptively with adjusted sample probabilities in order to mitigate biases. Finally, accuracies for images with high hair density and images with three different skin tones were measured for both classifiers. The results showed an increase in performance for images with darker skin tones.

In this work, we propose an adaptation of the bias mitigation method that can be used for more complex multi-class classification tasks. Also, our adaptation simplifies the process of experimenting with different architectures of the classifier. This is done by splitting the process into two steps. The initial step is to extract sample probabilities using a VAE which are no longer used adaptively. Instead, after obtaining the final values, a separate classification model is trained using those probabilities. Further, the process to extract sample probabilities from the latent space of a VAE is adapted to deal with multiple classes. This is done such that the probabilities are balanced with respect to all classes. When sampling an image, the probability of getting a specific class is equal for all classes, while the probabilities for individual images may still differ.

In our experiments, we build upon existing research and consider the attributes hairiness and skin tone. In addition to that, we consider the sex and age of the patient, which are not directly visually observable in the images.

We show that the adaptation is able to mitigate age and skin tone biases in a similar experiment setup as proposed by Das [Das21]. We use a similar dataset and a more advanced model architecture for the classifier and the VAE based on Residual Neural Networks (ResNets).

Further, we evaluate the effect of using transfer learning on the amount of bias and the ability to mitigate biases using the proposed method. We show that using transfer learning based on the popular ImageNet Large Scale Visual Recognition Challenge [Rus+15] decreases overall bias and increases the performance of the classifier. When additionally applying the bias mitigation method, we observe a further increase in overall performance. However, the method was not effective in significantly reducing biases.

Lastly, we apply our proposed method to a more complex multi-class classification task. Here, we can no longer observe an effect on overall performance or the ability to significantly reduce biases. We discuss potential reasons, which include having too few samples in order to learn a meaningful latent space from the dataset.

Overall, our results provide an adapted version of a bias mitigation method which succeeds to mitigate bias for a binary classification task. However, we were not able to provide evidence for the effectiveness of the method in more complex scenarios. Thus, further research is necessary to answer the question if the method can benefit the development of skin lesion classification algorithms for clinical processes.

In the following, we will introduce the methodology in detail and describe how the bias mitigation method was adapted. Next, we describe the overall experiment setup, followed by the individual experiments themselves. For every experiment, we evaluate and discuss the results. Lastly, we go into detail about the implications of our results and potential areas for future work.

2 Preliminaries

We consider the problem of multi-class image classification, where the task is to classify images from a domain X into one of the classes from a set Y. We say that $f_\theta \colon X \longrightarrow Y$ is a parametrized classifier where θ is a set of parameters. The goal is to optimize an initial set of parameters for the specific classification task at hand.

If the initial set of parameters is obtained from a related model trained on a different task, we call this transfer learning [PY10].

To optimize a parametrized classifier, we assume that we only have access to a limited number of images in X and their corresponding class labels. We define $D \subset X \times Y$ to be a dataset where $(x, y) \in D$ is a sample in which x is a representation of an image and y the corresponding class label.

An approach to making use of such a dataset is supervised learning, where the idea is to use known data to learn a classifier that generalizes to unseen data.

First, the parameters of f_θ are optimized to a training dataset D_{train}. During this process, which is called training, a loss function is minimized for all images in D_{train}. Instead of considering the final predictions p of the classifier, we take a look at intermediate class probabilities, which are denoted as $\hat{p}_i$ for all $i \in Y$. We consider the following loss, which is commonly used for classification tasks [Zha+21].

▶ **Definition 2.1.** Let x be an image, y its ground truth label, and n the number of classes. For all $i \in \{1, \dots, n\}$, let $\hat{p}_i$ be the corresponding predicted class probability for class i. The term,

$$loss_{CE}(x) = -\sum_{i=1}^{n} 1_{i=y} \log(\hat{p}_i)$$

is called cross-entropy loss, where 1 is the indicator function. ◀

In addition to D_{train}, a validation dataset D_{val} is used to adjust the hyperparameters of the training. A hyperparameter is, for instance, the learning rate, which determines how quickly or slowly the parameters are adjusted during the training. Adjusting hyperparameters based on a validation dataset can help to avoid issues such as overfitting, which is observed when a classifier performs very well on a training set but is unable to generalize to unseen data.

Finally, a third dataset D_{test} is provided to evaluate the ability of the classifier to classify unseen images from X. Let p be a vector of predictions and y a vector of ground truth labels. The commonly used metric accuracy [Zha+21] is defined as follows:

▶ **Definition 2.2.** The function

$$\mathrm{acc}(p, y) = \frac{1}{|p|} \cdot \sum_{i=1}^{|p|} 1_{p[i]=y[i]}$$

which takes a vector of predictions p and a vector of ground truth labels y defines the metric accuracy, where 1 is the indicator function and $|p|$ the length of the vector p. ◀

In other words, the metric accuracy is defined as the fraction of correctly classified images from a given dataset.

If the test dataset is highly imbalanced with respect to the classes, high accuracy does not necessarily correspond with a good classification ability. A classifier that always predicts the same class would achieve an accuracy of 90% for a dataset containing 90 samples from that class and 10 samples from a second class.

Therefore, we consider a slightly modified version of this metric, called weighted accuracy, in which each class's accuracy contributes equally to the final result. Let n be the number of classes, and for $1 \leq c \leq n$, let p_c be a vector of predictions for all samples (x, y) for which $y = c$. We denote y_c to be a vector of corresponding ground truth labels. Note that y_c is a vector in which all elements are c.

▶ **Definition 2.3.** The function

$$\mathrm{wacc}(p, y) = \sum_{i=1}^{n} \frac{1}{n} \mathrm{acc}(p_c, y_c)$$

which takes a vector of predictions p and a vector of ground truth labels y, defines the metric weighted accuracy. ◀

In addition to the classification performance, we evaluate if a classifier is biased with respect to an attribute of the images in the domain. We consider an attribute A that has $m \geq 2$ classes. For this attribute, we require that all images from the domain be categorized into one of the attribute classes.

We now define a notion of fairness following Amini et al. [Ami+19].

▶ **Definition 2.4.** We say that f_θ is a fair classifier with respect to an attribute A if the prediction accuracy is equal for all classes of the attribute. ◀

According to this definition, we measure the amount of bias with respect to an attribute by comparing the weighted accuracies of the classifier for the different classes of the attribute, as proposed by Amini et al. [Ami+19]. In other words, we calculate the variance of weighted accuracies for an attribute. We define wacc_i to be the weighted accuracy for the class i of attribute A measured on D_{test}. We denote $\overline{\mathrm{wacc}}$ as the average of all the classes weighted accuracies.

▶ **Definition 2.5.** For an attribute A with m classes, the term

$$\mathrm{bias}_A = \frac{1}{m}\sum_{i=1}^{m}(\mathrm{wacc}_i - \overline{\mathrm{wacc}})^2$$

defines the bias with respect to A. ◀

2.1 Using a ResNet as a Parametrized Classifier

For the experiments, we use a neural network as a parametrized classifier. The network takes an image with width w, height h, and number of channels c as input in the format of a $c \times h \times w$ matrix. For an image with three channels, the red, green, and blue values are provided for each pixel. The network processes this input according to its architecture and outputs a single number, which represents a class prediction. The weights of the individual layers of the network are the parameters that are optimized during the training process.

We use a neural network that follows the ResNet architecture, which was introduced by He et al. [He+16]. The fundamental idea behind ResNets is to include skip connections between layers so that the original input can reach deeper layers of the network more easily. Figure 2.1 shows an overview of a ResNet architecture consisting of 18 layers. The diagram illustrates the concept of skip connections for blocks of layers of the network. We refer the reader to chapter 8.6 of [Zha+21] for a detailed introduction and further background.

In comparison to standard CNNs, the ResNet architecture allows training much deeper networks effectively. When the architecture was introduced, it was able to win the ImageNet Large Scale Visual Recognition Challenge and had a big impact on the development of deep neural networks.

The architecture has been applied to skin cancer classification tasks by Yoon et al. [YHG19] while other architectures that build on the concept of having skip connections like DenseNet or ResNeXt have been explored by Gessert et al. [Ges+18].

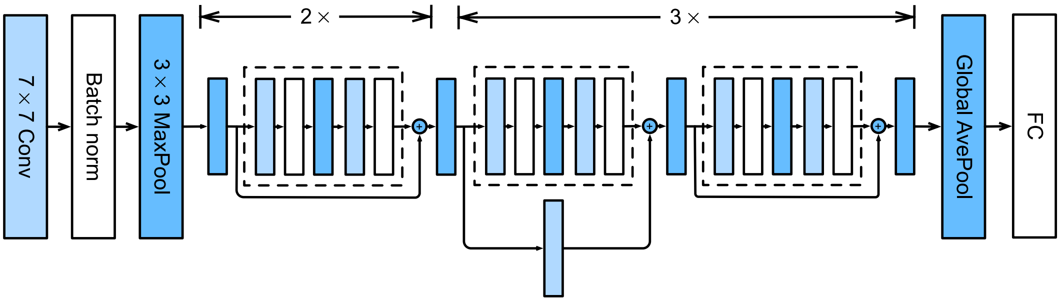


Figure 2.1: The diagram shows an overview of the architecture for a ResNet with 18 layers (ResNet18). The dashed lines mark residual blocks for which skip layer connections are added. Chapter 8.6 of [Zha+21] from which this figure is taken, provides more detailed background.

2.2 Variational Autoencoder for Bias Mitigation

In this section, we will introduce Variational Autoencoders (VAEs) which were proposed by Kingma and Welling [KW22] and explain how they provide a foundation to perform bias mitigation.

Given a dataset, a VAE can learn the underlying distribution, which is useful for many tasks. These include, for instance, generating similar data points like those from the training set and compressing high-dimensional data for subsequent processing or to reduce storage requirements [Roc21]. To mitigate bias, the distribution is used to identify samples with rare features in the dataset, which is described in detail in Chapter 3. Although VAEs can be used for all types of data, we will introduce them with visual data as an example, since this kind of VAE is used in our work.

To understand the concept behind VAEs, we will first introduce the ideas of dimensionality reduction and traditional autoencoders.

We define a function that takes an image as input and outputs values with a lower dimensionality as an encoder. The output of the encoder is considered to be the latent representation of the image, consisting of latent variables. The term latent space describes the set of all possible latent representations. Next, we define a function that takes a latent representation as input and outputs an image as a decoder. The output of the decoder has the same size as the input for the encoder. The process of decoding an image from its latent representation is called reconstruction.

These three elements, encoder, latent space, and decoder, define the architecture of an autoencoder, which can be seen in Figure 2.2. The goal of an autoencoder is to

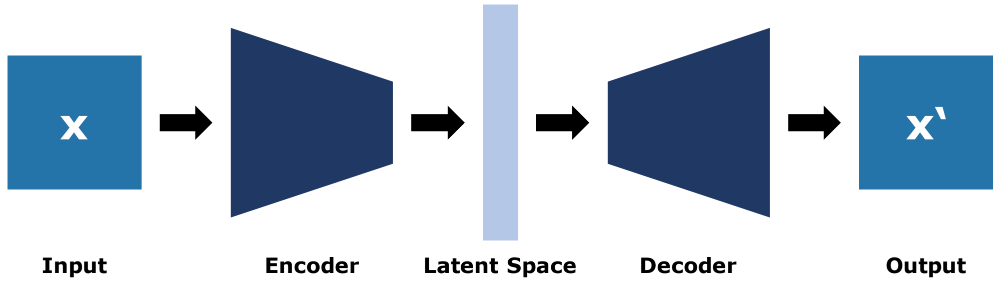


Figure 2.2: The architecture of an autoencoder has three main elements: The encoder, the latent space, and the decoder. For a VAE, the architecture is similar except for the latent space. A latent variable is no longer represented as a single value but with a mean μ and a standard deviation σ. To get a specific value for the latent variable, we sample from a normal distribution with these exact parameters.

reduce the dimensionality of the input image while keeping as much information as possible. The ability to reduce dimensionality is measured as the ability to reconstruct an image from its latent representation.

To evaluate the reconstruction ability, we use the metric Mean-Squared-Error which is commonly used as a loss term [GBC16].

▶ **Definition 2.6.** For an image x for which all values are flattened into a one dimensional vector with length n and its reconstruction x' of the same size, we define

$$loss_{MSE}(x) = \frac{1}{n}\sum_{i=1}^{n}(x_i - x_i')^2$$

to be the Mean-Squared-Error loss which we refer to as the reconstruction loss. ◀

An autoencoder is trained on a training dataset, and the reconstruction loss is measured on an unseen test dataset. The encoder and decoder of the autoencoder are symmetric to each other and can be implemented as a neural network. Common choices are basic CNNs or ResNets. The latter are described in more detail in Section 2.1.

A VAE is a special type of autoencoder that, instead of learning a specific latent representation for each image, learns a distribution of latent representations. This means that when encoding the image, the encoder does not output a single latent representation with a specific value for each latent variable. Instead, a latent variable is represented by a mean and a standard deviation. The latent representation is then sampled from the normal distributions defined by the means and standard deviations of the latent variables. The decoder takes the sampled latent representation as input and reconstructs the image.

As for the traditional autoencoder, the goal of a VAE is to minimize the reconstruction loss. At the same time, the difference between the distribution of every latent variable and a standard normal distribution is minimized. This second criterion ensures that the latent variables convey meaningful features of the images in the dataset. We measure this difference using the Kullback-Leibler divergence, which is defined as follows.

▶ **Definition 2.7.** For the probability distributions p and q, the term

$$\mathrm{KL}(p||q) = \int_{-\infty}^{\infty} p(x) \log \frac{p(x)}{q(x)} dx$$

is called the Kullback-Leibler divergence of p and q. The smaller the value of $\mathrm{KL}(p||q)$, the more similar the distributions. ◀

The loss for an image is the average Kullback-Leibler divergence of the distribution of a latent variable and a standard normal distribution.

▶ **Definition 2.8.** We consider a latent space consisting of d latent variables. For an image x and its latent representation, which is given as a vector of means μ and a vector of standard deviations σ each of length d, we define

$$loss_{KL}(x) = \frac{1}{d} \cdot \sum_{i=1}^{d} \mathrm{KL}(\mathcal{N}(\mu_i, \sigma_i)||\mathcal{N}(0,1))$$

to be the Kullback-Leibler divergence loss where $\mathcal{N}$ is a normal distribution with the respective parameters. ◀

The overall loss for a VAE is the sum of the reconstruction loss and the Kullback-Leibler divergence loss.

▶ **Definition 2.9.** For an image x, the term

$$loss(x) = loss_{MSE}(x) + \beta \cdot loss_{KL}(x)$$

defines the loss for x where $\beta \in [0,1]$ is a hyperparameter to balance the two loss terms. ◀

Once the VAE is trained to minimize this loss term, the latent space represents several features of the input data and their respective probability distributions. Note that these features are not necessarily human interpretable.

The idea to mitigate bias is now to use this learned latent space to determine the images in the training dataset that have rare features. These are then sampled more often during the training process in order to allow the model to better learn those rare features. This is described in detail in Chapter 3.

3 Adaptation of Bias Mitigation Method Using a VAE

In this chapter, we explain how we perform bias mitigation using the latent space of a VAE, which is an adaptation of the method introduced by Amini et al. [Ami+19]. We highlight the differences from the original method and explain the reasoning behind them. The overall idea is to learn a latent distribution of a training dataset with a VAE. This distribution is used to extract sample probabilities for the images in the training dataset, where images with rare features receive a higher probability. Then, these sample probabilities are used during the training of a classifier so that images with rare features are sampled more often.

3.1 Extracting Sample Probabilities From the Latent Space of a VAE

We assume to have a VAE and a dataset for which images we want to extract debiasing sample probabilities. We define m to be the number of latent variables of the VAE. Further, we denote n to be the number of classes contained in the dataset. We will consider each of those classes individually and extract sample probabilities such that they represent the distribution within that class. Afterward, we normalize the probabilities across all classes.

Consider a subset D_c of D that contains images from a single class c where $1 \leq c \leq n$. We now approximate the latent distribution for this class. Therefore, we calculate all the latent representations for the images in D_c using the encoder of the VAE. Remember that we receive for each of the m latent variables a mean μ and a standard deviation σ. Now, we only consider the means of the latent variables. For each latent variable, we create a histogram with ten bins. Figure 3.1 shows an example of such a histogram. Images that fall into a bin with lots of samples have a common expression of the feature, which is represented by the latent variable. Accordingly, bins with few samples contain images with rare expressions of that same feature.

The histograms of all m latent variables are then used to approximate the complete latent distribution for D_c and to finally calculate adjusted sample probabilities.

In [Ami+19] the exact process is not described in detail, which is why we follow the description of Das [Das21] where the method was applied.

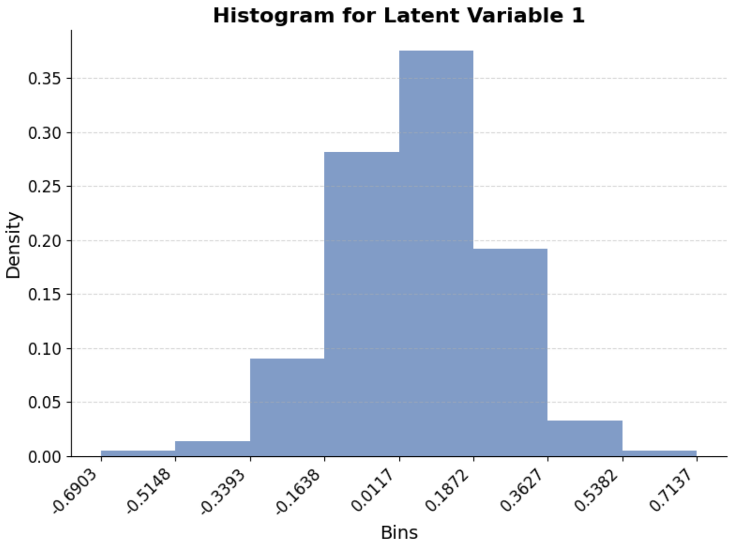


Figure 3.1: An intermediate step to extract debiasing sample probabilities is to calculate a histogram for each latent variable. Therefore, all images in a dataset are encoded into their latent representation. The histogram shows the distribution of means for the first latent variable. Images falling into a bin with lots of samples have a common expression of the feature represented by the latent variable. Accordingly, images falling into a bin with few samples have a rare expression of that feature.

First, we initialize an array that contains a probability for each image in D_c with zeros. Then, we perform the following steps for each latent variable $1 \leq i \leq m$:

1. A histogram for the latent variable i is created based on the means of all latent representations of images in D_c. An example of such a histogram is shown in Figure 3.1.

2. According to the distribution of the histogram, probabilities for each image are calculated, where images falling in bins with few samples receive a high probability.

3. For each image, we update the probability in the array if the newly calculated probability is higher than the current one.

As a last step, we normalize the probabilities to sum up to $\frac{1}{n}$ where n is the number of classes.

In [Ami+19], the method is described for a scenario with two classes where only the images of one of the classes receive adjusted sample probabilities. They also assume to have a balanced dataset with respect to the classes. Images from the second class all receive a sampling probability of $\frac{1}{|D| \cdot 2}$.

Our adaptation allows to have multiple classes. For each class, we calculate sample probabilities as described. By normalizing them to $\frac{1}{n}$ we ensure that adjusted sample probabilities sum up to 1 while not changing the property of the dataset to be balanced.

Note that despite having multiple classes, we only train a single VAE. That implies that the latent variables represent features that appear across all those classes. For our use case, this is acceptable, since we explicitly want to debias attributes that are independent of the specific class. In other use cases, it might be more appropriate to train a separate VAE for each of the classes in order to get a better performance.

3.2 Using Debiasing Sample Probabilities to Train a Classifier

After obtaining adjusted sample probabilities using the latent space of a VAE, we are now able to use them to train a classifier.

The original bias mitigation method by Amini et al. [Ami+19] incorporates the classifier into the encoder of the VAE. This allows to simultaneously train a VAE and the classifier. Each epoch, the sample probabilities are recalculated based on the updated VAE.

We propose to split the training of a VAE from the classification part. Thus, we first train a plain VAE and afterward calculate adjusted sample probabilities once. Then, these are used to train a classifier where the final probabilities are used in every epoch.

In comparison to adaptively using provisional sample probabilities, relying on final values for the whole training potentially increases the ability to mitigate bias.

Apart from that, the split has two major advantages. First, training a VAE is more time-consuming than training a classifier, which follows the same architecture as the encoder of the VAE. One reason for that is the increased number of parameters. Thus, the split can be beneficial when optimizing hyperparameters for the classifier, for which the training is repeated many times. Second, the architecture of the classifier is no longer bound to the VAE. This allows to try out different models without having to adjust the architecture of the VAE.

A minor disadvantage of this adaptation is that performing one complete training takes longer since two models need to be trained.

4 Experimental Setup

We performed several experiments to test the proposed bias mitigation method. These experiments all share a common setup, which is described in this chapter. Overall, we train a classifier with and without bias mitigation in three different configurations and compare the results.

The underlying problem is the task of skin lesion classification based on images, where the domain X would consist of images of potential skin lesions.

4.1 Dataset and Additional Metadata

We use the dataset for task three of the ISIC 2018 Challenge [Cod+19; TRK18]. The dataset contains over 10,000 images of different skin lesions that have been collected over several years.

The images are labeled with one of seven skin lesions, which are abbreviated with *MEL*, *NV*, *BCC*, *AKIEC*, *BKL*, *VASC*, and *DF*. Appendix A provides some further explanation on these classes. We perform experiments containing two classes (*MEL*, and *NV*) as well as four classes (*MEL*, *NV*, *BKL*, and *BCC*). The dataset is limited to those classes respectively.

Additionally, for every image, some metadata is provided by the ISIC Archive and can be obtained using their API [ISI23b]. The metadata includes the approximate age and the sex of the patient from whom the image was obtained. For our experiments, we further enrich this metadata with information about the hairiness of the skin and the skin tone. Details on that are described below.

We follow the split into training, validation, and test datasets that is proposed by the challenge.

4.1.1 Dealing with High Class Imbalances

The seven classes of the dataset are highly imbalanced. Table 4.1 shows an overview of the number of samples per class for the training dataset. Note that more than half of the images are from one single class. When training a classifier on a dataset with high class imbalances, the classifier will tend to predict the most common class more often. To overcome this, we limit the number of samples per class in the training set so that all classes have the same number of samples. This approach to dealing with imbalanced data is commonly called undersampling [MRA20].

Having an unbalanced test or validation dataset is not a problem if used with a metric such as weighted accuracy, which takes the class imbalances into account.

Skin Lesion	Samples in the Train Dataset
MEL	**1113**
NV	**6705**
BCC	**514**
BKL	**1099**
AKIEC	327
DF	115
VASC	142

Table 4.1: The table shows the number of samples per class in the training dataset. The four most represented classes are highlighted. When performing undersampling for these classes, the number of samples per class is limited to 514.

4.1.2 Detecting Skin Tone Based on Fitzpatrick Scale

We use the approach proposed by Das [Das21] to enhance the dataset with information about the skin tone based on the Fitzpatrick Scale [GS19]. The images are classified into Type I, Type II, or Type III of the scale. This is done by blurring the image and applying different filters corresponding to the types. The image is stored in the BGR format, which means that every pixel is represented by three values: blue (b), green (g) and red (r). The filters are applied to the blue and green values of the image and use the following thresholds:

- Type I: $180 < b < 255$ and $180 < g < 255$
- Type II: $150 < b < 200$ and $150 < g < 200$
- Type III: $80 < b < 150$ and $80 < b < 150$

The filter with the fewest black pixels is selected as the skin tone of the image. See Figure 4.1 for an example. In this case, the skin was classified as Type I on the Fitzpatrick Scale.

4.1.3 Measuring Presence of Visible Hair in Images

We enhance the dataset with information about the hairiness of the images. Therefore, we follow an adapted version of the approach proposed by Das [Das21].

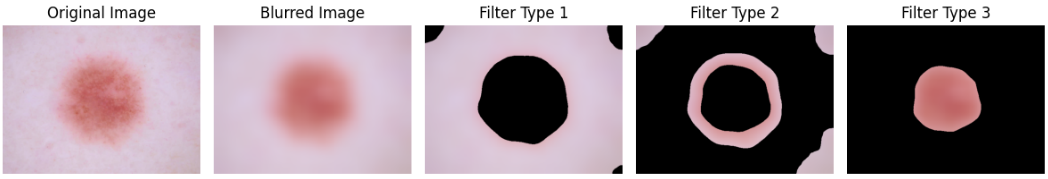


Figure 4.1: These steps are performed to determine the skin tone. After blurring the image, several filters are applied. The corresponding skin tone of the filter that leads to the fewest black pixels is selected. In this case, the image is classified as showing skin with a skin tone of Type I on the Fitzpatrick Scale.

Using the original method, we observe that there are only two images showing high hair density in the test dataset for the class *MEL*. These are too few samples to evaluate the bias mitigation ability. Therefore, instead of labeling images with high hair density, we consider all images that contain visible hair. In that way, we get more samples in that class while still having an interesting attribute to evaluate bias.

The adapted procedure includes the following steps:

1. The image is converted to grayscale and blurred.
2. Next, a black-hat morphological transformation is applied to the image.
3. The remaining highlighted pixels are considered hair. If more than 2% of the pixels are highlighted, the image is considered to include visible hair.
4. Positive results are checked manually and corrected if necessary.

Figure 4.2 shows an example of the hair detection process, where the last image correctly indicates the presence of hair. In contrast to Das [Das21], we use a lower threshold of 2% instead of 7% to detect hair and additionally perform a manual check afterward. For the four most represented classes of the test dataset, the number of samples per class and the respective number of images with visible hair are shown in Table 4.2.

Skin Lesion	Samples in the Test Dataset	Samples with Visible Hair
MEL	171	18
NV	909	105
BCC	93	14
BKL	217	27

Table 4.2: The table shows the number of samples per class in the test dataset and the number of images containing visible hair for the respective class.

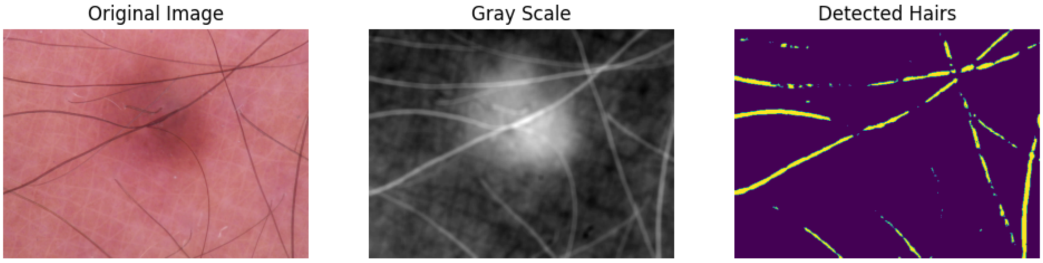


Figure 4.2: To detect if an image contains visible hair, the original image is blurred and converted into grayscale. Then, a black-hat morphological transformation is applied. Based on a threshold, we decide if the image contains hair or not.

4.1.4 Transformations and Dataset Normalization During Training

We apply dataset normalization as a standard preprocessing step for training the ResNet model in image classification [GBC16]. We calculate the mean and the standard deviation for the dataset and use these to normalize the images.

Additionally, we perform other transformations during the training of the classifiers as well as the training of VAEs. The specific transformations are selected based on manual tests of different configurations using a ResNet without debiasing for the binary classification task. We selected the configuration that achieved the highest weighted accuracy on the validation dataset. Overall, the images, which were originally 450 × 600 are transformed as follows:

1. We crop the image to a square of 450 × 450.
2. We perform a resized crop to a size of 360 × 360.
3. During the training process, we flip the image horizontally with a probability of 50% every time the image is sampled.
4. We normalize the image according to the mean and standard deviation of the dataset.

4.2 Attributes for Bias Evaluation

In our experiments, we evaluate the bias of the classifiers for several attributes. In particular, we consider the attributes age and sex of the patient, as well as visible hair and skin tone based on the Fitzpatrick Scale. For the attribute age of the patient, we evaluate the following age groups:

- Up to age 30
- Age 31 up to age 55
- Age 56 or older

The attribute sex is evaluated using the classes male and female, while the attribute visible hair is separated into a class with images containing visible hair and a class with images not containing visible hair. The process of extracting information about the presence of hair is described in detail in Subsection 4.1.3.

For the last attribute, skin tone, we consider Type I, Type II, and Type III of the Fitzpatrick Scale as the classes for this attribute. Subsection 4.1.2 describes the process of determining the skin tone from the images.

For all these attributes, bias is measured by calculating the variance of weighted accuracies between the classes of the respective attribute.

To check if we can observe significant bias reduction, we rely on the Mann-Whitney U test, which is a non-parametric statistical test to check if two underlying distributions are equal. We use a two-sided Mann-Whitney U test since we are also interested in determining if a bias has increased significantly due to the application of the bias mitigation method. Since we evaluate bias for four attributes and have three different configurations for our experiments, we perform twelve statistical tests in total. Performing multiple tests increases the likelihood of obtaining false-positive results. Therefore, we apply the Bonferroni correction and choose an adjusted significance level of $\alpha = \frac{0.05}{12} = 0.0042$ [Hol79].

4.3 Training of the Classifiers

For all experiments, we use the same kind of parametrized classifier that follows the ResNet architecture with 18 layers, as described in Section 2.1. The weights of the model represent the parameters. We train the classifier with a learning rate of 10^{-5} for 20 epochs. We chose the learning rate based on manuel tests using a ResNet without performing debiasing for a binary classification task. Several values between 10^{-3} and 10^{-6} were tested. A learning rate of 10^{-5} achieved the best weighted accuracy on the validation dataset. We limited the number of epochs to 20 since we observed that a larger number of epochs did not improve the weighted accuracy on the training dataset. Based on the weights of the last epoch, we evaluate the performance of the classifier using the metric weighted accuracy. We use this metric since the mentioned ISIC challenge [Cod+19] used weighted accuracy as the primary metric to rank submissions.

For the experiments, we adjust the number of classes as well as the configuration of the initial parameter set. Additionally, we adjust the sample probabilities when performing debiasing.

4.4 Training of a VAE to Extract Sample Probabilities

According to the bias mitigation method described in Chapter 3, we train a VAE and extract adjusted sample probabilities using the latent space.

For the encoder network, we use a ResNet with 18 layers. The latent space consists of 256 latent variables. For the decoder, we use a network symmetric to the ResNet of the encoder. The specific implementation was derived from PyTorch Lightning Bolts [Bol23]. We followed the proposed default of 256 for the number of latent variables and the default learning rate of 10^{-4}.

We train the VAE for 200 epochs. This number of epochs allows the model to converge well without leading to a very long training time. The Kullback-Leibler divergence loss is weighted with a factor of $\beta = 0.1$, which is again the proposed default of Bolts [Bol23]. We monitor the best model using the validation dataset. This means we consider the model with the lowest validation loss for the final evaluation on the test dataset.

We will now give an example of the VAE in order to show how it is able to provide sample probabilities that distinguish between rare and common samples of the dataset. For this, we use a VAE that is trained for the experiment with two classes.

We calculated sample probabilities as described in Chapter 3. Figure 4.3 shows the histogram of calculated probabilities. One can see that lots of images have a low sample probability, but the number of images decreases as the sample probability increases.

If we visualize images with low and high sample probabilities, we indeed observe an increase in the diversity of images with higher sample probabilities, as can be seen in Figure 4.4.

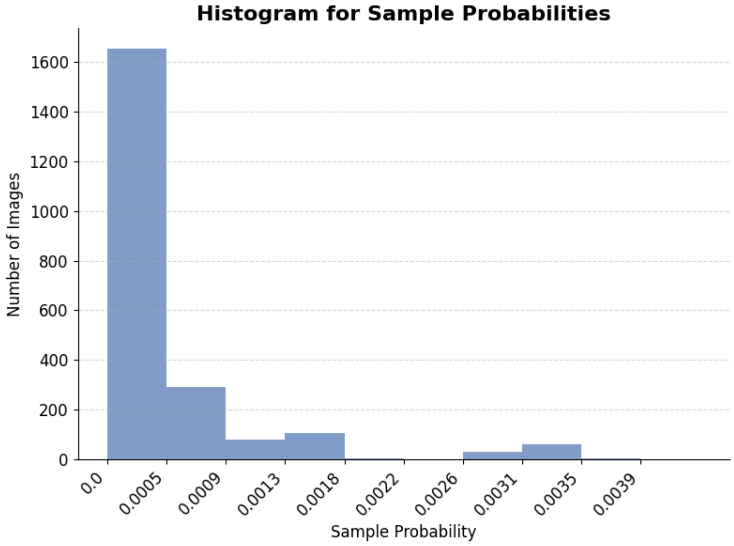
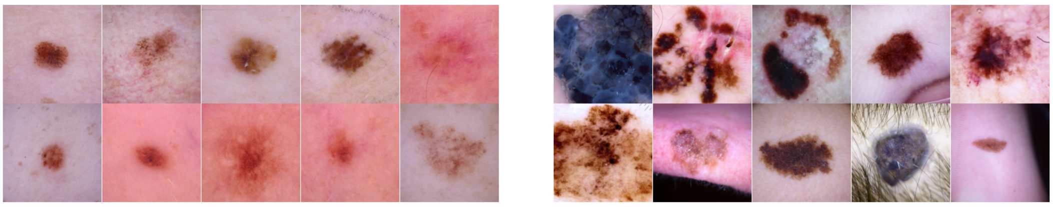


Figure 4.3: The figure shows a histogram of sample probabilities for a training dataset with two classes. The sample probabilities are extracted from the latent space of a VAE. We observe that the number of images decreases with increasing probabilities.



(a) Top ten images with the lowest sample probability.

(b) Top ten images with the highest sample probability.

Figure 4.4: The figures show images with high and low sample probabilities, respectively. We observe that the images with a high sample probability are more diverse. This suggests that the method is able to distinguish between images with common and rare features.

5 Bias Mitigation for Binary Classification Task

With the following experiment, we show that our proposed adaptation of the bias mitigation method works in a similar setup as the one in [Das21]. We compare the two results in detail. The problem considered is a binary skin lesion classification task with the classes Melanoma (MEL) and Melanocytic Nevus (NV). We trained a VAE to extract debiasing sample probabilities from the latent space once. The VAE achieved an overall loss of 0.2014 which is composed of a reconstruction loss of 0.176, and a Kullback-Leibler divergence of 0.02543. Then, we trained a classifier based on the ResNet18 architecture with and without debiasing for ten times, respectively.

5.1 Improved Overall Performance

We observe that the debiasing method is able to improve the overall weighted accuracy, while also improving the weighted accuracy for every single class of the considered attributes. Figure 5.1 shows an overview of these findings. The overall weighted accuracy improved on average by 6.98% from 71.69% to 78.67%.

5.2 Mitigated Biases

The bias mitigation method significantly mitigated biases for the attributes age and skin tone. When taking a look at the weighted accuracies for the different classes of the attributes age in Figure 5.1, we observe that the differences between the age groups are smaller with debiasing in place. The same is observable for the three skin tone classes, suggesting that bias was mitigated for these two attributes.

In Figure 5.2 we show the amount of bias with and without debiasing for all considered attributes. Here, we can confirm our observation. The bias for the attribute age was reduced on average from 0.023 to 0.012 while the bias for the attribute skin tone was reduced on average from 0.020 to 0.010. We observe that the attribute sex has very little bias in the first place. This finding can be explained by the fact that the dataset was almost evenly balanced with respect to the attribute classes male and female. Thus, we did not expect the classifier to have much bias with respect to this attribute.

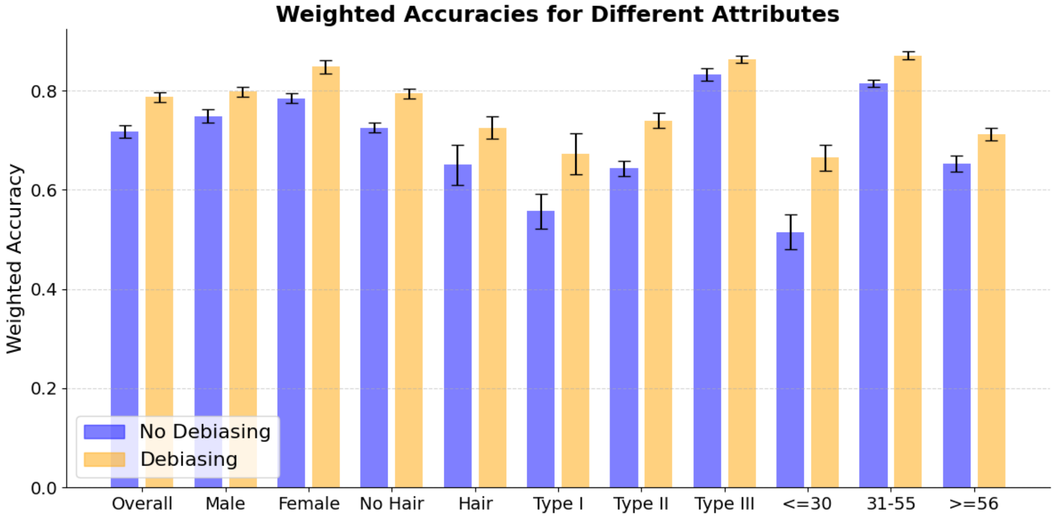


Figure 5.1: The diagram shows the weighted accuracies of a classifier for a binary classification task with and without debiasing for several attributes. The classifiers were trained ten times each. We observe that the overall weighted accuracy increased on average by 6.98% with debiasing in place.

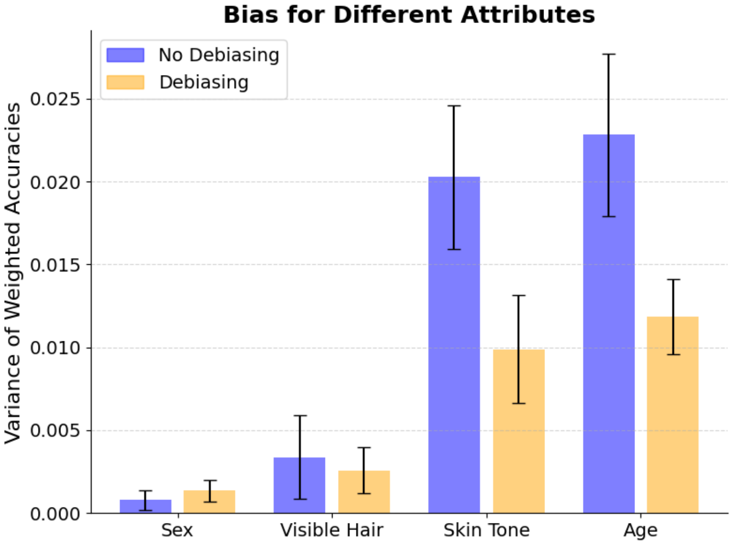


Figure 5.2: The diagram shows the biases of a classifier for a binary classification task, with and without debiasing for several attributes. The classifiers were trained ten times each. We observe that biases for the attributes age and skin tone decreased by more than 45% on average.

For all attributes, we conducted a Mann-Whitney U test with a corrected significance level of 0.0042 to check if the observed difference is significant. An overview of the results of the tests is shown in Table 5.1. The biases with respect to the sex of the patient and the attribute visible hair did not change significantly. We observe a significant effect on the biases with respect to skin tone and age.

Attribute	p-value
Sex	0.07566
Visible Hair	0.79134
Skin Tone	0.00044
Age	0.00025

Table 5.1: We performed a Mann-Whitney U test for each attribute to check if the amount of bias changed significantly. We observe a significant decrease in bias for the attributes skin tone and age.

5.3 Comparison to Results of Related Work

In the work of Das [Das21] a similar binary classification task was considered while using a slightly different dataset. Instead of distinguishing between Melanoma and Melanocytic Nevus, they considered Melanoma and a second class combined of all benign skin lesions in the ISIC2018 dataset. They evaluated their results based on the attributes skin tone and high hair density. However, they did not measure bias as proposed by Amini et al. [Ami+19]. Instead, they only calculated accuracies for all classes of the attributes. These were compared for classifiers with and without debiasing in place.

The evaluation was done using a test dataset consisting of 1,113 images of melanoma and 31,958 images of other skin lesions.

Note that the dataset is highly imbalanced and that they used the metric accuracy without weighting it with respect to the classes. Therefore, it is questionable if the reported improved accuracy for images having a skin tone of Type III on the Fitzpatrick Scale results in a better ability to detect melanoma on images of this attribute class. However, they provided AUC scores, which are another metric that indeed accounts for an imbalanced test dataset. These results suggest that classification ability was improved for images with skin tones of Type II or Type III.

In our experiment, the AUC scores as shown in Figure 5.3 only suggest a marginal improvement for images showing no hair and images of Type I or Type II. Thus, we do not observe a similar effect on the AUC scores.

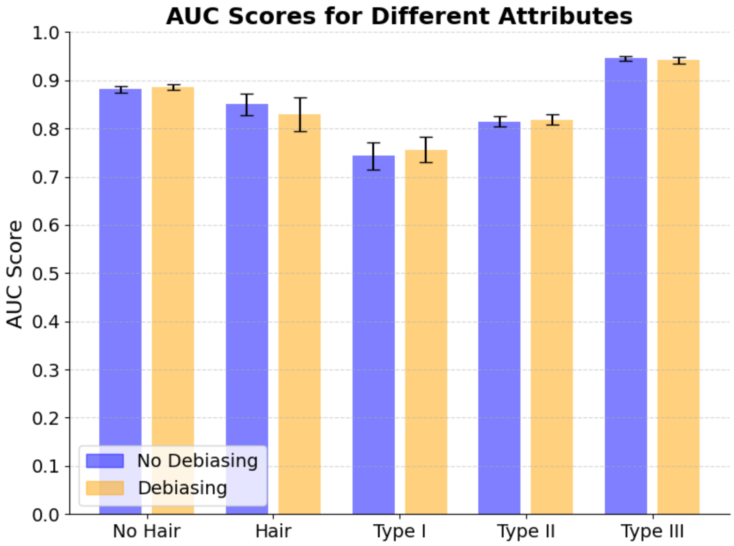


Figure 5.3: The diagram shows the AUC scores of a classifier for a binary classification task, with and without debiasing for the attributes visible hair and skin tone. The classifiers were trained ten times each. We observe a marginal improvement for images showing no hair and images of Type I or Type II on the Fitzpatrick Scale.

Nonetheless, we observe an increased classification ability based on the metric weighted accuracy for all classes of the attributes visible hair and skin tone. Additionally, our adaptation of the bias mitigation method significantly mitigates bias with respect to skin tone. Since Das [Das21] did not include a metric for the exact amount of bias, we are unable to compare the methods in that regard.

Therefore, performing further experiments using both versions would help to better compare the two approaches.

6 Effects of Using Transfer Learning on Bias Mitigation

In this chapter, we describe an experiment to measure the effect of using transfer learning on the ability to mitigate biases. Transfer learning showed to be successful for skin lesion classification [HKF20]. Therefore, this approach is likely to be applied to classifiers that could potentially be used in clinical processes.

We trained a ResNet18 for a binary classification task with the classes Melanoma (*MEL*) and Melanocytic Nevus (*NV*) with pretrained weights from the ImageNet Large Scale Visual Recognition Challenge [Rus+15]. Then, we repeated the experiment with bias mitigation in place, for which we used the same sample probabilities as in Chapter 5. Both versions were trained 10 times for better statistical significance.

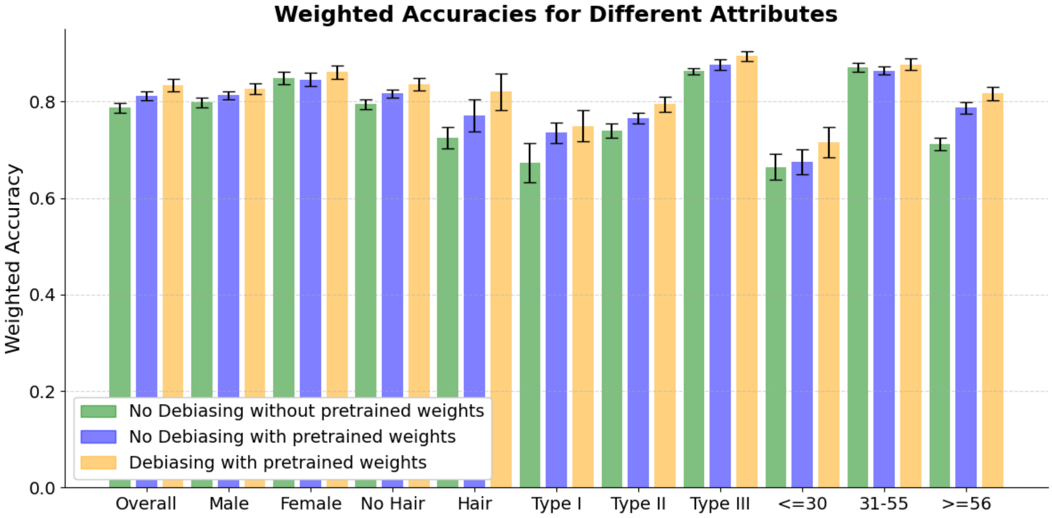


Figure 6.1: The diagram shows the weighted accuracies of a classifier for a binary classification task. The accuracies are shown for three different configurations of the classifier, which was trained ten times for every configuration. We observe that using transfer learning increases overall and attribute-based weighted accuracy. Additionally performing debiasing further increases the overall weighted accuracy by 2.3% on average.

We observe that using pretrained weights reduces biases on its own. However, when additionally applying bias mitigation, we no longer observe a debiasing

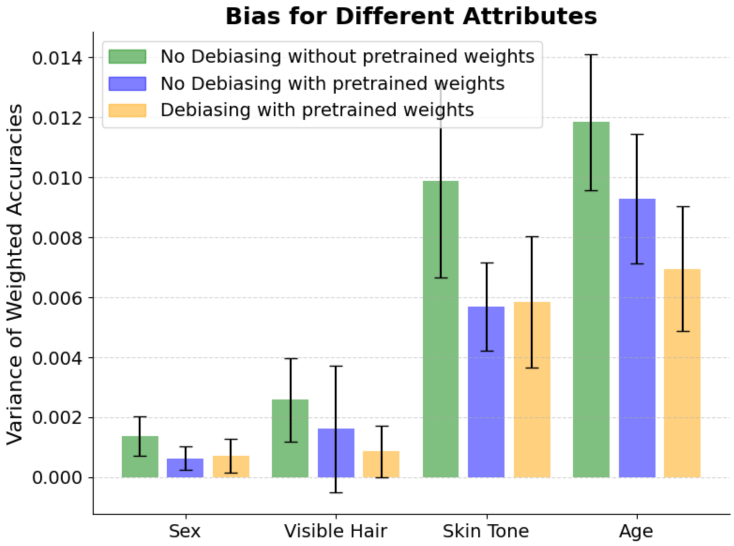


Figure 6.2: The diagram shows the biases of a classifier for a binary classification task. The classifier was trained in three different configurations for ten times each. We observe that using transfer learning decreases biases for all attributes except sex. Additionally performing debiasing has no significant effect on the amount of bias.

effect. Figure 6.1 shows the overall as well as attribute-based weighted accuracies for the two versions and a version without pretrained weights from the previous experiment. In comparison to not using pretrained weights, the overall weighted accuracy improved by 9.4% from 71.7% to 81.1%. When applying bias mitigation, we observe a further improvement of 2.3% to 83.4%. Also, all accuracies for the classes of the considered attributes increased with bias mitigation in place. However, the difference is less significant than in the previous experiment without transfer learning.

When measuring biases for all the attributes as shown in Figure 6.2, we observe that the biases are already low without debiasing. For instance, the bias with respect to skin tone is at 0.0057 while it is at 0.020 when not using pretrained weights. We can no longer observe significant bias mitigation on any of the attributes. We performed a Mann-Whitney U test for all attributes, which further supports this observation, as can be seen in Table 6.1.

These results suggest that applying transfer learning with pretrained weights from ImageNet can improve overall performance and reduce biases on its own. However, the proposed bias mitigation method is no longer effective.

Attribute	p-value
Sex	0.96985
Visible Hair	0.67758
Skin Tone	0.90972
Age	0.05390

Table 6.1: We performed a Mann-Whitney U test for each attribute to check if the amount of bias changed significantly. We observe no significant change in any of the attributes. This suggests that the bias mitigation method is not effective in conjunction with transfer learning.

7 Bias Mitigation for Multi-Class Classification Task

In this experiment, we consider a multi-class classification task with four classes that consist of Melanoma (*MEL*), Melanocytic Nevus (*NV*), Benign Cell Carcinoma (*BCC*) and Benign Keratosis (*BKL*). We trained a VAE on a balanced dataset with the four classes and extracted debiasing sample probabilities. The VAE reached an overall loss of 0.2229 while having a reconstruction loss of 0.1985 and a Kullback-Leibler divergence loss of 0.02437. Then, we trained a classifier based on the ResNet18 architecture with and without debiasing for ten times, respectively.

As shown in Figure 7.1 we do not see a significant difference in overall weighted accuracy. Also, the accuracies for the different classes of the considered attributes did not change as much as in the previous experiments. This already suggests that the method is not working as well as it did for the less complex task of binary classification.

We measured bias for all considered attributes and performed a Mann-Whitney U test with a corrected significance level of 0.0042 to check if the bias significantly changed. Figure 7.2 shows the biases, while Table 7.1 contains the results of the statistical test.

We observe that bias with respect to sex was very low in the first place and did not change with bias mitigation in place. Since the training dataset was balanced with respect to sex, this is what we expected. Bias with respect to the attribute visible hair was reduced on average from 0.0057 to 0.0033 while bias with respect to the age of the patient was reduced from 0.0430 to 0.0388. However, these differences

Attribute	p-value
Sex	0.57061
Visible Hair	0.00729
Skin Tone	0.03121
Age	0.08897

Table 7.1: We performed a Mann-Whitney U test for each attribute to check if the amount of bias changed significantly. We observe no significant change in any of the attributes. Thus, the bias mitigation method was not effective in this experiment for a multi-class classification task.

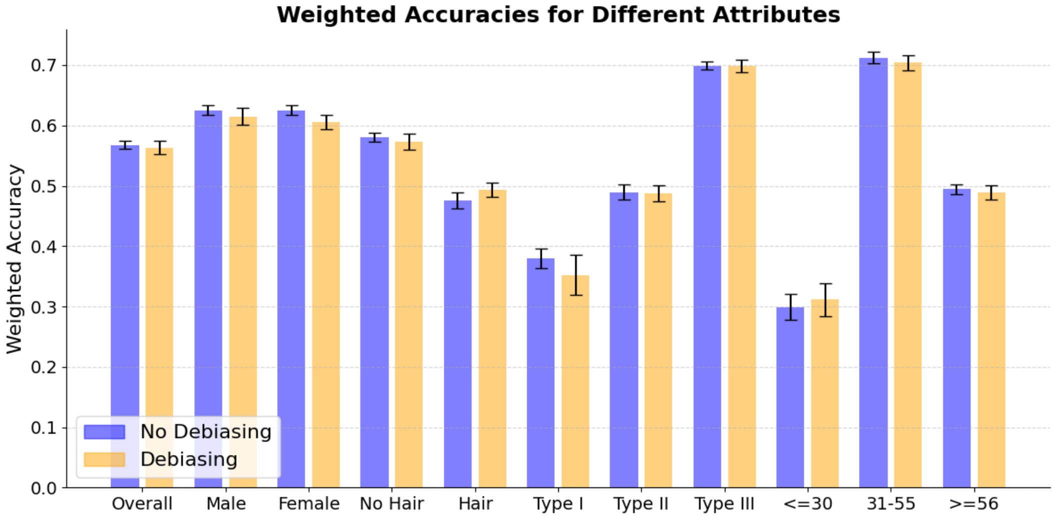


Figure 7.1: The diagram shows the weighted accuracies of a classifier for a multi-class classification task with and without debiasing for several attributes. The classifiers were trained ten times each. We observe no significant effect of debiasing on the overall weighted accuracy.

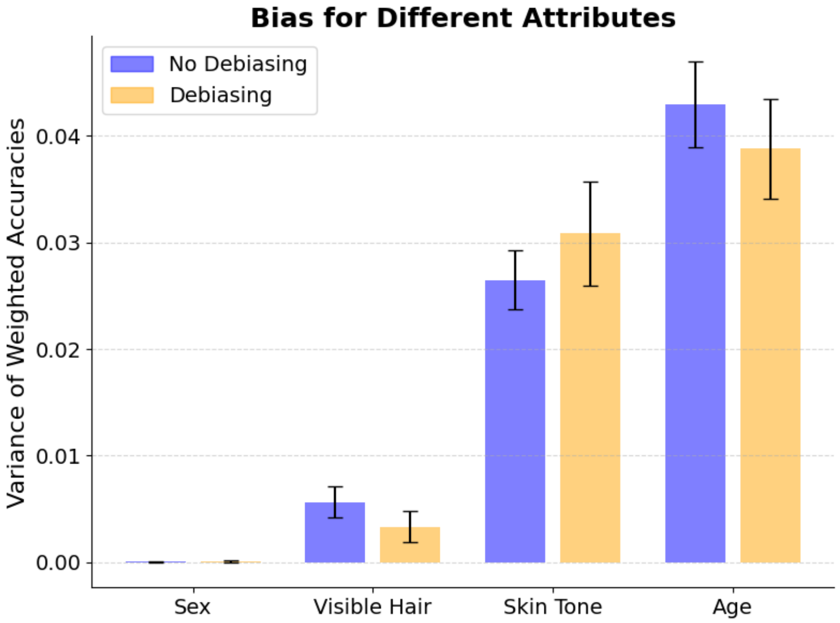


Figure 7.2: The diagram shows the biases of a classifier for a multi-class classification task, with and without debiasing for several attributes. The classifiers were trained ten times each. We observe that the bias mitigation method was not effective.

are not significant. For the attribute skin tone, we observe an increase in bias with mitigation in place, which is again not significant.

Overall, the method was not effective for this multi-class classification task. We discuss potential reasons and areas for future work in Chapter 8.

8 Discussion and Future Work

In this work, we propose an adaptation of a bias mitigation method based on the learned latent space of VAEs. Our adaptation allows to use this method with multiple classes and makes it easier to experiment with different architectures for parametrized classifiers by splitting the process into two steps.

Overall, we addressed whether the method can help to reduce biases in skin lesion classification models in order to make use of them in clinical scenarios. Specifically, we measure bias with respect to the sex, age and skin tone of the patient, as well as the attribute visible hair.

We were able to show that our suggested adaptation is able to mitigate biases for a similar setup as it was done with the original method. For a binary skin lesion classification task, the method decreased bias with respect to the age and skin tone of the patient. Bias with respect to age was reduced from 0.023 to 0.012 on average. A Mann-Whitney U test confirmed this decrease with a p-value of 0.00025. Significant bias reduction for the attribute skin tone occurred on average from 0.020 to 0.010 with a p-value of 0.00044.

For the same task, we showed that using transfer learning is able to improve overall performance while reducing biases in comparison to classifiers without pretrained weights. However, with transfer learning in place, the bias mitigation method is no longer as effective, showing no significant bias reduction.

Lastly, we applied the method to a more complex multi-class classification task. Here, we could not observe any increase in overall performance nor a reduction in bias. A potential reason for this could be that the used dataset is too small, since we perform undersampling to deal with the otherwise high class imbalance. The ability of the VAE to learn relevant latent variables increases with the size of the provided dataset. Thus, the learned latent space of our VAE might not provide enough meaningful information to effectively calculate debiasing adjusted sample probabilities. Another reason would be that the images of the classes are visually too different to learn a valuable common latent space. Despite having in common that they contain skin lesions, their characteristics vary. Therefore, it is promising to use a separate VAE for each of the classes. This would potentially increase the ability to reduce biases based on the specific characteristics of the particular skin lesions.

Apart from that, future work should investigate if using a different approach to deal with the high class imbalances can increase the debiasing ability. Promising methods could include oversampling or balancing the weights accordingly. Further, one could investigate if doing thorough hyperparameter tuning is effective. Relevant hyperparameters include, for instance, the number of latent variables of the VAE, the factor with which the reconstruction loss is weighted, and general hyperparameters such as the learning rate or the number of epochs.

Bibliography

[AAW20] Adel Abusitta, Esma Aïmeur, and Omar Abdel Wahab. **Generative Adversarial Networks for Mitigating Biases in Machine Learning Systems**. en. *Santiago de Compostela* (2020). URL: https://ecai2020.eu/papers/348_paper.pdf (see page 2).

[Ami+19] Alexander Amini, Ava P. Soleimany, Wilko Schwarting, Sangeeta N. Bhatia, and Daniela Rus. **Uncovering and Mitigating Algorithmic Bias through Learned Latent Structure**. In: *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*. AIES '19. New York, NY, USA: Association for Computing Machinery, Jan. 2019, 289–295. ISBN: 978-1-4503-6324-2. DOI: 10.1145/3306618.3314243 (see pages 2, 6, 7, 13, 15, 27).

[BG18] Joy Buolamwini and Timnit Gebru. **Gender Shades: Intersectional Accuracy Disparities in Commercial Gender Classification**. en. In: *Proceedings of the 1st Conference on Fairness, Accountability and Transparency*. ISSN: 2640-3498. PMLR, Jan. 2018, 77–91. URL: https://proceedings.mlr.press/v81/buolamwini18a.html (visited on 06/28/2023) (see page 1).

[Bol23] PyTorch Lightning Bolts. *PyTorch Lightning Bolts*. 2023. URL: https://www.pytorchlightning.ai/bolts (visited on 06/26/2023) (see page 22).

[Cam+17] Alex Campolo, Madelyn Rose Sanfilippo, Meredith Whittaker, and Kate Crawford. **AI Now 2017 Report**. AI Now Institute at New York University, 2017. URL: https://ainowinstitute.org/publication/ai-now-2017-report-2 (visited on 06/28/2023) (see page 1).

[CK19] L. Elisa Celis and Vijay Keswani. *Improved Adversarial Learning for Fair Classification*. en. arXiv:1901.10443 [cs, stat]. Jan. 2019. URL: http://arxiv.org/abs/1901.10443 (visited on 06/18/2023) (see page 2).

[Cod+19] Noel Codella, Veronica Rotemberg, Philipp Tschandl, M. Emre Celebi, Stephen Dusza, David Gutman, Brian Helba, Aadi Kalloo, Konstantinos Liopyris, Michael Marchetti, Harald Kittler, and Allan Halpern. *Skin Lesion Analysis Toward Melanoma Detection 2018: A Challenge Hosted by the International Skin Imaging Collaboration (ISIC)*. arXiv:1902.03368 [cs]. Mar. 2019. DOI: 10.48550/arXiv.1902.03368 (see pages 1, 17, 22, 45).

[Com+22] Marc Combalia, Noel Codella, Veronica Rotemberg, Cristina Carrera, Stephen Dusza, David Gutman, Brian Helba, Harald Kittler, Nicholas R. Kurtansky, Konstantinos Liopyris, Michael A. Marchetti, Sebastian Podlipnik, Susana Puig, Christoph Rinner, Philipp Tschandl, Jochen Weber, Allan Halpern, and Josep Malvehy. **Validation of artificial intelligence prediction models for skin cancer diagnosis using dermoscopy images: the 2019 International Skin Imaging Collaboration Grand Challenge**. English. *The Lancet Digital Health* 4:5 (May 2022). Publisher: Elsevier, e330–e339. ISSN: 2589-7500. DOI: 10.1016/S2589-7500(22)00021-8 (see page 1).

[Das21] Sauman Das. **Automated Bias Reduction in Deep Learning Based Melanoma Diagnosis using a Semi-Supervised Algorithm**. In: *2021 IEEE International Conference on Bioinformatics and Biomedicine (BIBM).* Dec. 2021, 1719–1726. DOI: 10.1109/BIBM52615.2021.9669772 (see pages 2, 3, 13, 18, 19, 25, 27, 28).

[GBC16] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. **Deep Learning**. MIT Press, 2016. URL: http://www.deeplearningbook.org (visited on 06/28/2023) (see pages 9, 20).

[Ges+18] Nils Gessert, Thilo Sentker, Frederic Madesta, Rüdiger Schmitz, Helge Kniep, Ivo Baltruschat, René Werner, and Alexander Schlaefer. *Skin Lesion Diagnosis using Ensembles, Unscaled Multi-Crop Evaluation and Loss Weighting*. en. arXiv:1808.01694 [cs]. Aug. 2018. URL: http://arxiv.org/abs/1808.01694 (visited on 06/15/2023) (see page 7).

[GS19] Vishal Gupta and Vinod Kumar Sharma. **Skin typing: Fitzpatrick grading and others**. *Clinics in Dermatology* 37:5 (2019). The Color of Skin, 430–436. ISSN: 0738-081X. DOI: https://doi.org/10.1016/j.clindermatol.2019.07.010 (see page 18).

[He+16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. **Deep Residual Learning for Image Recognition**. In: June 2016, 770–778. DOI: 10.1109/CVPR.2016.90 (see page 7).

[HKF20] Khalid M. Hosny, Mohamed A. Kassem, and Mohamed M. Fouad. **Classification of Skin Lesions into Seven Classes Using Transfer Learning with AlexNet**. *Journal of Digital Imaging* 33:5 (Oct. 2020), 1325–1334. ISSN: 0897-1889. DOI: 10.1007/s10278-020-00371-9 (see page 29).

[Hol79] Sture Holm. **A Simple Sequentially Rejective Multiple Test Procedure**. *Scandinavian Journal of Statistics* 6:2 (1979), 65–70. ISSN: 03036898, 14679469. URL: http://www.jstor.org/stable/4615733 (visited on 06/27/2023) (see page 21).

[ISI23a] International Skin Image Collaboration ISIC. *About*. en. 2023. URL: https://www.isic-archive.com/mission (visited on 06/18/2023) (see page 1).

[ISI23b] International Skin Image Collaboration ISIC. *ISIC Archive*. 2023. URL: https://api.isic-archive.com/api/docs/swagger/ (visited on 05/12/2023) (see page 17).

[KW22] Diederik P. Kingma and Max Welling. *Auto-Encoding Variational Bayes*. Dec. 2022. DOI: 10.48550/arXiv.1312.6114 (see page 8).

[MRA20] Roweida Mohammed, Jumanah Rawashdeh, and Malak Abdullah. **Machine Learning with Oversampling and Undersampling Techniques: Overview Study and Experimental Results**. In: *2020 11th International Conference on Information and Communication Systems (ICICS)*. ISSN: 2573-3346. Apr. 2020, 243–248. DOI: 10.1109/ICICS49469.2020.239556 (see page 18).

[PY10] Sinno Jialin Pan and Qiang Yang. **A Survey on Transfer Learning**. *IEEE Transactions on Knowledge and Data Engineering* 22:10 (Oct. 2010). Conference Name: IEEE Transactions on Knowledge and Data Engineering, 1345–1359. ISSN: 1558-2191. DOI: 10.1109/TKDE.2009.191 (see page 5).

[Roc21] Joseph Rocca. *Understanding Variational Autoencoders (VAEs)*. en. Mar. 2021. URL: https://towardsdatascience.com/understanding-variational-autoencoders-vaes-f70510919f73 (visited on 04/28/2023) (see page 8).

[Rus+15] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. **ImageNet Large Scale Visual Recognition Challenge**. en. *International Journal of Computer Vision* 115:3 (Dec. 2015), 211–252. ISSN: 1573-1405. DOI: 10.1007/s11263-015-0816-y (see pages 3, 29).

[Soc23] American Cancer Society. *Melanoma Survival Rates | Melanoma Survival Statistics*. en. 2023. URL: https://www.cancer.org/cancer/types/melanoma-skin-cancer/detection-diagnosis-staging/survival-rates-for-melanoma-skin-cancer-by-stage.html (visited on 05/25/2023) (see page 1).

[TRK18] Philipp Tschandl, Cliff Rosendahl, and Harald Kittler. **The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions**. en. *Scientific Data* 5:1 (Aug. 2018), 180161. ISSN: 2052-4463. DOI: 10.1038/sdata.2018.161 (see pages 17, 45).

[WCR22] World Cancer Research Fund WCRF. *Skin cancer statistics | World Cancer Research Fund International*. en-US. 2022. URL: https://www.wcrf.org/cancer-trends/skin-cancer-statistics/ (visited on 05/25/2023) (see page 1).

[WHO17] World Health Organisation WHO. *Radiation: Ultraviolet (UV) radiation and skin cancer*. en. 2017. URL: https://www.who.int/news-room/questions-and-answers/item/radiation-ultraviolet-(uv)-radiation-and-skin-cancer (visited on 05/25/2023) (see page 1).

[Xu+18] Depeng Xu, Shuhan Yuan, Lu Zhang, and Xintao Wu. **FairGAN: Fairness-aware Generative Adversarial Networks**. In: Dec. 2018, 570–575. DOI: 10.1109/BigData.2018.8622525 (see page 2).

[YHG19] Chris Yoon, Ghassan Hamarneh, and Rafeef Garbi. **Generalizable Feature Learning in the Presence of Data Bias and Domain Class Imbalance with Application to Skin Lesion Classification**. en. In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2019*. Ed. by Dinggang Shen, Tianming Liu, Terry M. Peters, Lawrence H. Staib, Caroline Essert, Sean Zhou, Pew-Thian Yap, and Ali Khan. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2019, 365–373. ISBN: 978-3-030-32251-9. DOI: 10.1007/978-3-030-32251-9_40 (see page 7).

[Zha+21] Aston Zhang, Zachary C. Lipton, Mu Li, and Alexander J. Smola. **Dive into Deep Learning**. *arXiv preprint arXiv:2106.11342* (2021). URL: https://d2l.ai/ (visited on 06/28/2023) (see pages 5–8).

I hereby declare that this thesis is my own unaided work. All direct or indirect sources used are acknowledged as references.

Potsdam, June 29, 2023

Jacob Schäfer

A Appendix

Skin lesion	Short Explanation
Melanoma (MEL)	"Melanoma is a malignant neoplasm derived from melanocytes that may appear in different variants."
Melanocytic Nevus (NV)	"Melanocytic nevi are benign neoplasms of melanocytes and appear in a myriad of variants."
Basal Cell Carcinoma (BCC)	"Basal cell carcinoma is a common variant of epithelial skin cancer that rarely metastasizes but grows destructively if untreated. It appears in different morphologic variants (flat, nodular, pigmented, cystic)."
Actinic Keratosis and Intraepithelial Carcinoma (AKIEC)	"Actinic Keratoses (Solar Keratoses) and Intraepithelial Carcinoma (Bowen's disease) are common noninvasive, variants of squamous cell carcinoma."
Benign Keratosis (BKL)	"'Benign keratosis' is a generic class that includes seborrheic keratoses ('senile wart'), solar lentigo - which can be regarded a flat variant of seborrheic keratosis - and lichen-planus like keratoses (LPLK), which corresponds to a seborrheic keratosis or a solar lentigo with inflammation and regression."
Vascular Lesion (VASC)	"Vascular skin lesions in the dataset range from cherry angiomas to angiokeratomas and pyogenic granulomas. Hemorrhage is also included in this category."
Dermatofibroma (DF)	"Dermatofibroma is a benign skin lesion regarded as either a benign proliferation or an inflammatory reaction to minimal trauma."

Table A.1: We use the dataset for task three of the ISIC 2018 challenge [Cod+19; TRK18]. The table provides short explanations for the different classes of the dataset. The explanations are taken from Tschandl et al. [TRK18].