COMP 472

Artificial Intelligence

Summer 2022

Group : PG_09

Christopher Cui 40096627 - Compliance Specialist

Sami Ibrahim 40156134 - Data Specialist

Ibrahim Ibrahim 40158162 - Training Specialist

Usama Saleem 40110036 - Evaluation Specialist

Project Repository : https://github.com/chriscui47/Project-1-cnn

Network model trained on biased data:

https://drive.google.com/file/d/1pEUcMMn-wUoHObN6JY-j4XDHC3RchXNT/view?usp=sharing

Network model trained on unbiased data:

https://drive.google.com/file/d/18lq8K65bGajkH5v8qLKR-jR0C79htCHw/view?usp=sharing

Unbiased Data:

https://drive.google.com/drive/folders/1ICDeMX8gu_QkYYRbcxBkxZdhrHmGdX6n?usp=sharing

Dataset

We primarily built our dataset using two sources which had around 5000 images each. The first source [1] is from a public dataset provided by Humans In the loop, which is “a social enterprise that aims to connect conflict-affected communities to digital work”. The dataset consists of 6k images acquired from the public domain with an extreme attention to diversity, featuring people of all ethnicities, ages, and regions. The second source [2] is from the FaceMask database, where images were “manually collected from Google Images and annotated using Labelling tool in YOLO format”. The Database consists of 4,866 images of people in different environments and situations. We then selected 1600 images, and manually split the images into four different classes by putting them into a separate folder for each class: Person without a face mask (400 images), Person with a “community” (cloth) face mask (400 image), Person with a “surgical” (procedural) mask (400 images), and Person with a “FFP2/N95/KN95”-type mask (300 images). The testing data consisted of 100 images from each class. The validation data was split from the initial 1600 images in a 80/20 ratio, where 80% of the images were used for training and the remaining 20% was used as validation data.

The resolution of the images varied from 450x325 pixels all the way up to 6240x4160 pixels. However, the images were all resized to 150x150 as part of our pre-processing. Furthermore, our pre-processing also converted the images to tensors, and then normalized the images. Our normalization consisted of using the transform. Normalize function, and setting both the mean and standard deviation to 0.5. This ends up normalizing the image to the range [-1,1]. This reduces the skewness and helps the CNN learn faster.

CNN Architecture

Overview

The basis of this architecture has “hidden layers” called convolutional layers, which receives input, transforms the input in some way, and outputs the transformed input to the next layer. In this architecture, this transformation is a convolution operation, and each layer has filters to detect patterns, which get more accurate as the layers progress and increase.

In our architecture, we have 3 CNN Blocks, with each block consisting of 2 convolutional layers and 1 max-pooling layer. In the max pooling layer, we reduce the size of the image to half of the original size before passing it onto the next layer.

We are creating a class that inherits from the nn.Module to define different layers of the network. The first step is to use the nn.Sequential module to create sequentially ordered layers in the network. Each consists of a convolution + ReLU + pooling sequence. In each convolution layer, use LeakyRelu for the activation function and BatchNorm2d to accelerate the training process. [4]

ReLU

As an activation function, the ReLU function was used to prevent the negative values in the output from the input of the network and achieve a non-linear transformation of the data. If the output is negative, then the output is set to zero, and if the output is positive, then the output is set to the original value. The function is $f(x) = \max(0,x)$ and the ReLU is used to speed up training in the neural network. In addition, we hope for ReLU transformation in hopes that the transformed data will be close to linear (regression) or close to linearly separable (classification). In essence, the ReLU used is a compact and efficient way of moving signals from between the layers, resulting in faster computation and a more efficient CNN.

Epoch

We have an epoch of 12, which is the number of times the network is trained. Each epoch refers to when the entire dataset completes one propagation forward and backwards through the network. [3].

If we increase the number of epochs, the accuracy of the network will increase, although with a cost of time and computational power, thus a good balance of 12 is what we have set on.

FC Layer

In our architecture, we implement a fully connected layer which is used to connect the output of the convolutional layer to the fully connected layer. The output of the convolutional layer is a vector size of 82944 and as for the fully connected layer, that size is 1024. In essence, a FC layer is nothing more than a dense network of neurons and the connections between two neurons. We use this to classify an image to a specific category, for instance a mask type, after we have extracted the features from the image using CNN. Once we find the features of an image, we flatten the features found into a 1D layer, which is then used as an input to the FC layer. In the end, the number of neurons resulting in the final output after all the layers will be correlated to the number of categories of classification we are aiming for. In our case, we end up with 4 final neurons. Not only does the FC layer categorize the features along the way, it learns to associate features to a specific label, thus making it an essential part of our CNN.

Skorch

The goal of skorch is to make it possible to use PyTorch with scikit-learn. This is achieved by providing a wrapper around PyTorch that has a scikitlearn interface. Additionally, skorch abstracts away the training loop, a simple net.fit(X, y) is enough. We are using net.fit(train_data, y=y_train)to train our network. Skorch also takes advantage of scikit-learn functions, such as predict. We are using y_pred = net.predict(test_data)to predict the test data.

skorch.NeuralNetClassifier is a Neural Network class used for classification tasks. We Initialize the NeuralNetClassifier class then train the CNN model using the method fit. [4]

Evaluation for Part 1

Based on the confusion matrix, labels 2 and 3 have relatively good accuracy. The big issue arises when trying to predict cloth masks and n-95/ffp2 masks. There may be a bit of accuracy skewness because the dataset is not completely balanced for label 1 (n95 masks). This is because we have around 75 less training images than the other classes. We recognize this issue and will focus on rectifying this during the second phase of the project. Another thing we must do for the next part of the project is to improve the accuracy which currently sits at 0.46 which is not optimal at all. This can be accomplished by adding more training data (especially N95 class, which is our lowest accuracy).

Another thing we will do is to add more layers to our CNN architecture. With more layers, the neural net would be able to more fine tune the results from more calculations. Specifically, we will aim to increase the number of hidden layers which may increase the accuracy of the results.

K-Fold Cross-Validation

For part 1 of our report, we did not k-fold cross validate our model, however we did for this part.

In terms of our results, we have accuracy measurements for each fold as so:

10-Fold cross validation for the original data, part 1 data:

K-Fold	Accuracy
1	0.61864407
2	0.64957265
3	0.76923077
4	0.69230769
5	0.72649573
6	0.72649573
7	0.78632479
8	0.75213675
9	0.52991453
10	0.7008547
—----------	—----------
AVERAGE	**0.695197741**

We utilized Skorch to compute the k-fold. As we notice, the average accuracy across all 10 folds is respectable at 69%. However, we see differences in terms of the categories of biases we chose. First, we will look at gender. In terms of the female category, we noticed a significant drop in accuracy at 54%:

10-Fold cross validation for only female:

K	Accuracy
1	0.58536585
2	0.625
3	0.425
4	0.475
5	0.625
6	0.6
7	0.525
8	0.65
9	0.45
10	0.525
—----------	—----------
AVERAGE	**0.548536585**

It was significantly harder for our neural net to detect masks efficiently.

Here are all the data points like above, for male-only, dark-skin-only, and light-skin-only, with a discussion about them at the end:

10-Fold cross validation for only male:

K-Fold	Accuracy
1	0.51219512
2	0.45
3	0.525
4	0.55
5	0.6
6	0.5
7	0.5
8	0.5
9	0.525
10	0.6
—----------	—----------
AVERAGE	**0.526219512**

Compared to male-data, we notice a 2.2% increase in accuracy for the female subset, and this is due to the fact that we had more female data points than male. In the end, the network was more capable of detecting masks on female candidates than their counterpart.

10-Fold cross validation for only dark skin:

K-Fold	Accuracy
1	0.61290323
2	0.61290323
3	0.64516129
4	0.41935484
5	0.67741935
6	0.64516129
7	0.58064516
8	0.53333333
9	0.56666667
10	0.56666667
—----------	—----------
AVERAGE	**0.586021506**

10-Fold cross validation for only light skin:

K-Fold	Accuracy
1	0.65714286
2	0.51428571
3	0.68571429
4	0.62857143
5	0.68571429
6	0.68571429
7	0.73529412
8	0.64705882
9	0.73529412
10	0.67647059
—----------	—----------
AVERAGE	**0.665126052**

As we can evidently see, the accuracy for lighter-skinned individuals was significantly higher, at almost 10% greater. Unlike gender, this result does not result from lack of equal data, but rather the lighter skin is easier for the neural network to distinguish a mask and it's type. In a camera for instance, the more light we have, the more details it can distinguish, including from our own eyes. Thus it isn't a stretch to say that our neural network can detect masks easier with lighter skinned individuals. There is clear evidence in bias towards lighter-skinned individuals, and to rectify this, more data on the darker side is needed to train the neural net further, as it seemed easier for the network to distinguish feature sets in the light skin sub-category.

In order to help resolve this bias, we normalized the ratio of pictures of each sub category to be equal and retrained the dataset. The reduction of the bias is detailed at the end of this report. After this, the results for each k-fold were as so:

10-Fold cross validation for only original, part 1 data (unbias):

K-Fold	Accuracy
1	0.61864407
2	0.64957265
3	0.76923077
4	0.69230769
5	0.72649573
6	0.72649573
7	0.78632479
8	0.75213675
9	0.52991453
10	0.7008547
—----------	—----------
AVERAGE	**0.695197741**

10-Fold cross validation for only female (unbias):

K	Accuracy
1	0.51219512
2	0.75
3	0.4
4	0.55
5	0.625
6	0.6
7	0.575
8	0.6
9	0.5
10	0.55
—----------	—----------
AVERAGE	**0.566219512**

10-Fold cross validation for only male (unbias):

K-Fold	Accuracy
1	0.53658537
2	0.5
3	0.575
4	0.525
5	0.675
6	0.575
7	0.45
8	0.625
9	0.475
10	0.55
—----------	—----------
AVERAGE	**0.548658537**

10-Fold cross validation for only dark skin (unbias):

K-Fold	Accuracy
1	0.67741935
2	0.48387097
3	0.41935484
4	0.35483871
5	0.51612903
6	0.48387097
7	0.61290323
8	0.46666667
9	0.53333333
10	0.56666667
—----------	—----------
AVERAGE	**0.511505377**

10-Fold cross validation for only light skin (unbias):

K-Fold	Accuracy
1	0.68571429
2	0.62857143
3	0.85714286
4	0.71428571
5	0.6
6	0.74285714
7	0.64705882
8	0.73529412
9	0.70588235
10	0.70588235
—----------	—----------
AVERAGE	**0.702268907**

The distribution of the data used to train the model before adjusting for bias is shown in the table below.

Original biased data (Gender):

	Female		Male	
	826		746	
	Dark female	Light female	Dark male	Light male
No mask	89	66	144	106
Cloth	96	153	68	95
N95	91	97	117	85
Surgical	117	117	62	69

==========

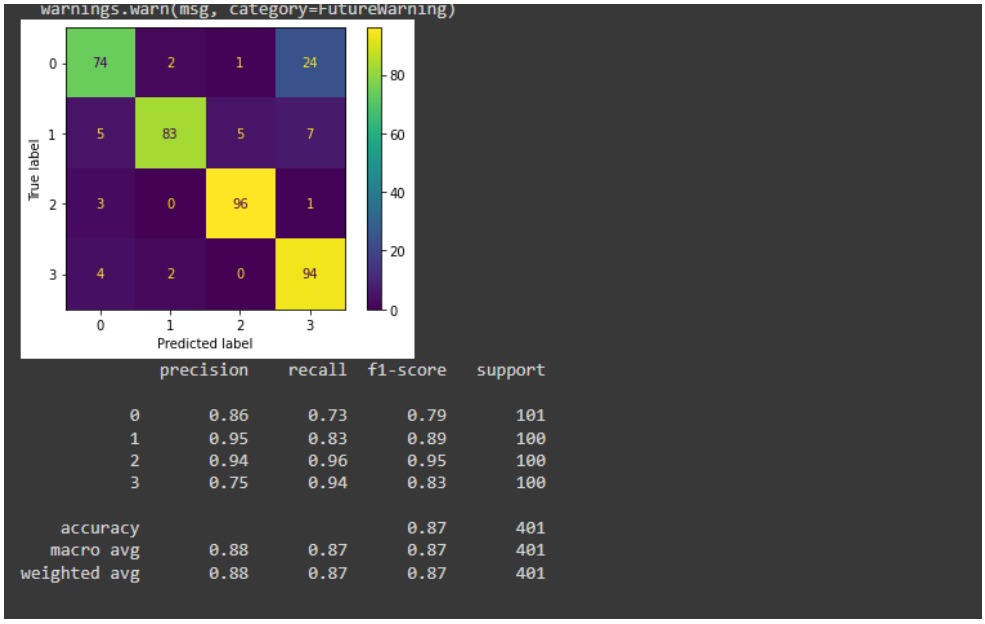
Original biased data (Race):

	Dark		Light	
	784		788	
	Dark female	Dark male	Light female	Light male
No mask	89	144	66	106
Cloth	96	68	153	95
N95	91	117	97	85
Surgical	117	62	117	69

Evaluation from Original Model (Part 1)

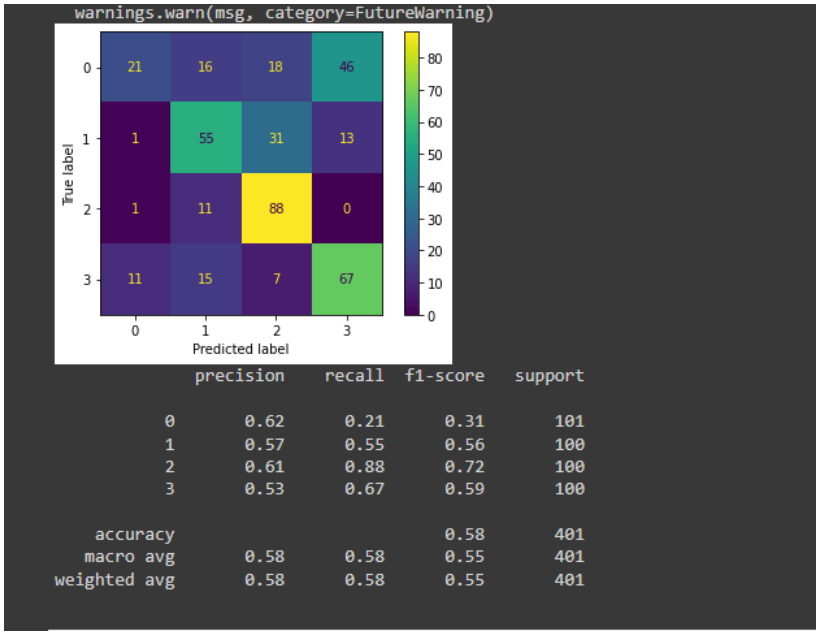
The following are evaluation metrics for each category before we fixed the bias in our model:

Testing on Female only (biased data)



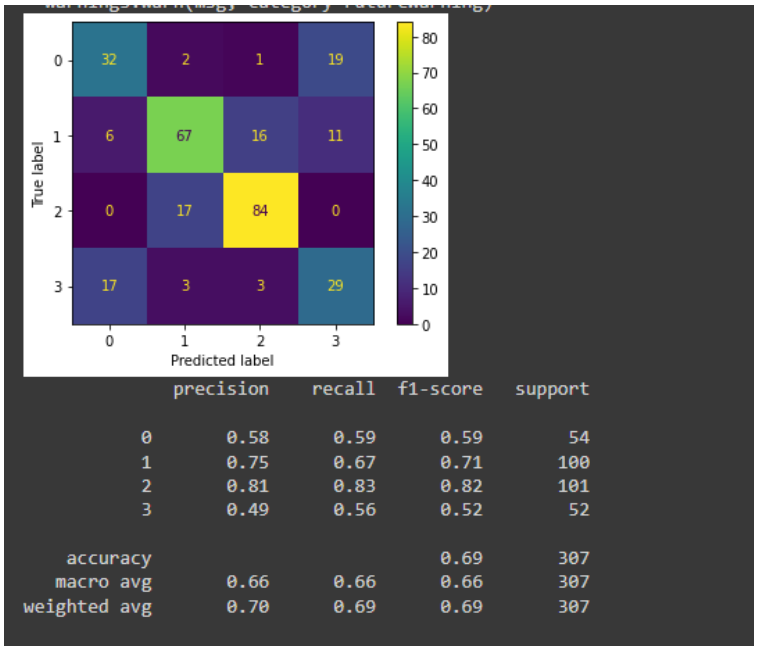
	precision	recall	f1-score	support
0	0.86	0.73	0.79	101
1	0.95	0.83	0.89	100
2	0.94	0.96	0.95	100
3	0.75	0.94	0.83	100
accuracy			0.87	401
macro avg	0.88	0.87	0.87	401
weighted avg	0.88	0.87	0.87	401

Testing on Male only (biased data)



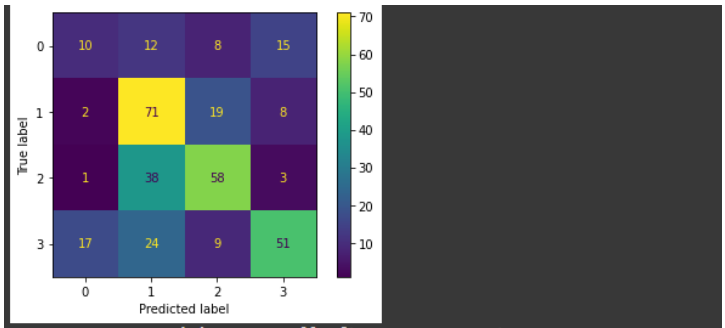
	precision	recall	f1-score	support
0	0.62	0.21	0.31	101
1	0.57	0.55	0.56	100
2	0.61	0.88	0.72	100
3	0.53	0.67	0.59	100
accuracy			0.58	401
macro avg	0.58	0.58	0.55	401
weighted avg	0.58	0.58	0.55	401

Testing on dark skin only (biased data)



	precision	recall	f1-score	support
0	0.58	0.59	0.59	54
1	0.75	0.67	0.71	100
2	0.81	0.83	0.82	101
3	0.49	0.56	0.52	52
accuracy			0.69	307
macro avg	0.66	0.66	0.66	307
weighted avg	0.70	0.69	0.69	307

Testing on light skin only (biased data)



	precision	recall	f1-score	support
0	0.33	0.22	0.27	45
1	0.49	0.71	0.58	100
2	0.62	0.58	0.60	100
3	0.66	0.50	0.57	101
accuracy			0.55	346
macro avg	0.53	0.50	0.50	346
weighted avg	0.56	0.55	0.54	346

We notice a higher accuracy of testing on dark skinned individuals (0.69) compared to light skinned (0.55). This bias can be introduced in our model due to having more pictures of dark skinned individuals that our model was trained on. Specifically, we have

784 dark skinned images and 788 light skinned images as training data before rectifying the bias. The main steps we took to remove the bias of our model was to introduce more light skinned images with the goal of improving the accuracy of that subclass as well as increase the number of female data points to make the ratio of all data more or less equal to give the network a fair chance at reducing as much bias as possible.

Evaluation of the unbiased model:

New unbiased data

	868 Female		**868 Male**	
New unbiased data	**Dark female**	**Light female**	**Dark male**	**Light male**
No mask	117	117	117	117
Cloth	100	100	100	100
N95	100	100	100	100
Surgical	117	117	117	117

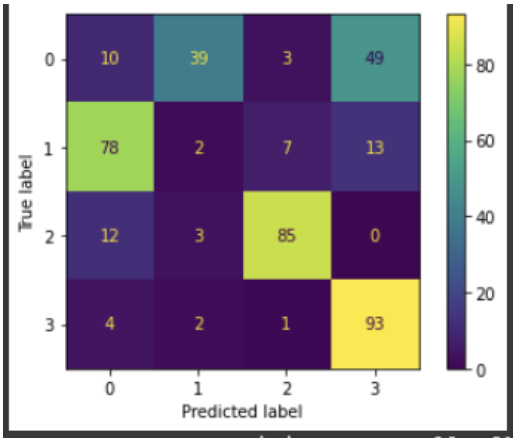
New unbiased data

	868 Dark		**868 Light**	
New unbiased data	**Light male**	**Light female**	**Dark male**	**Dark female**
No mask	117	117	117	117
Cloth	100	100	100	100
N95	100	100	100	100
Surgical	117	117	117	117

In order to help resolve this bias, we normalized the ratio of pictures of each sub category to be equal and retrained the dataset as shown in the above table.

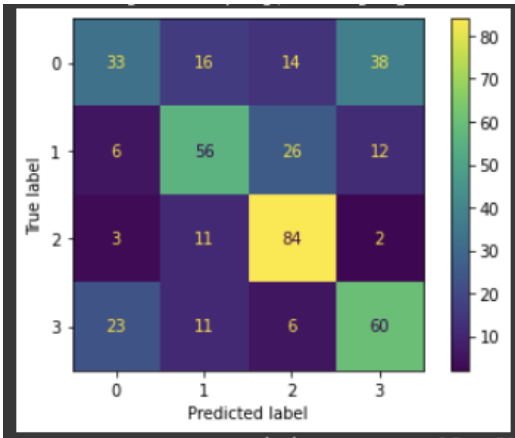
After resolving this bias, the results are as shown:

Testing on Female only (unbiased)



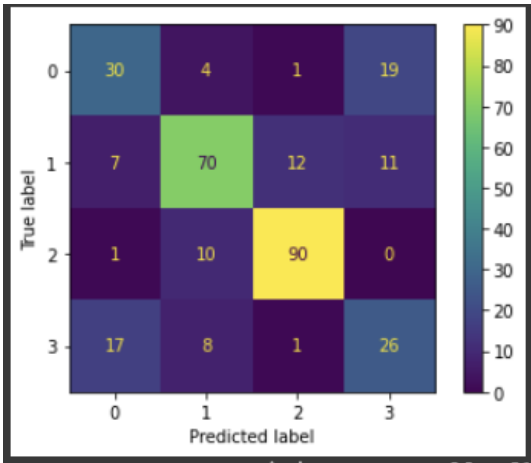
	precision	recall	f1-score	support
0	0.10	0.10	0.10	101
1	0.04	0.02	0.03	100
2	0.89	0.85	0.87	100
3	0.60	0.93	0.73	100
accuracy			0.47	401
macro avg	0.41	0.47	0.43	401
weighted avg	0.41	0.47	0.43	401

Testing on Male only (unbiased)



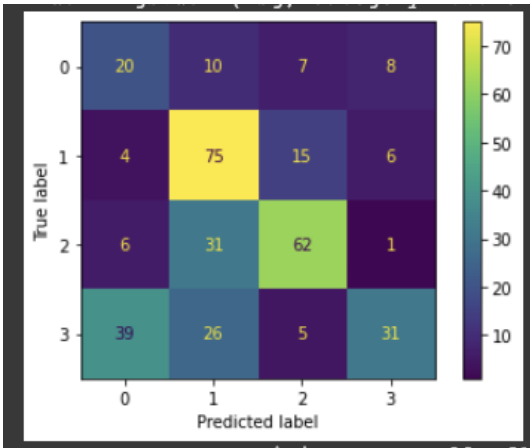
	precision	recall	f1-score	support
0	0.51	0.33	0.40	101
1	0.60	0.56	0.58	100
2	0.65	0.84	0.73	100
3	0.54	0.60	0.57	100
accuracy			0.58	401
macro avg	0.57	0.58	0.57	401
weighted avg	0.57	0.58	0.57	401

Testing on dark skin only (unbiased)



	precision	recall	f1-score	support
0	0.55	0.56	0.55	54
1	0.76	0.70	0.73	100
2	0.87	0.89	0.88	101
3	0.46	0.50	0.48	52
accuracy			0.70	307
macro avg	0.66	0.66	0.66	307
weighted avg	0.71	0.70	0.70	307

Testing on light skin only (unbiased)



	precision	recall	f1-score	support
0	0.29	0.44	0.35	45
1	0.53	0.75	0.62	100
2	0.70	0.62	0.66	100
3	0.67	0.31	0.42	101
accuracy			0.54	346
macro avg	0.55	0.53	0.51	346
weighted avg	0.59	0.54	0.54	346

Bias Detection & Analysis

The attributes selected to determine bias were race and gender. Given the vast amount of data, race was divided by darker skinned individuals, as well as lighter skinned individuals. This binary approach allowed us to clearly see any bias in our AI implementation, if any, given the distinct differences in skin color. As for gender, we are also using a binary approach of female and male. After resolving the bias, in this part we determined higher accuracy as expected.

Gender

In terms of the female dataset, after applying the k-fold we saw an increase in accuracy. The mean accuracy increased from 54% to 57%. In terms of male candidates, the accuracy was 54% versus the female accuracy of 57%. As for the male aspect specifically, the accuracy previous to the bias elimination was a mean of 52% whereas after resolving the network and k-fold, it was 54% which is nonetheless an increase in accuracy and reduction in bias across gender. Comparatively to part 1, we see an increase in accuracy by ~56% throughout the entire network through f-fold cross validation.

Race

In terms of the race dataset, we have compared light skinned versus dark skinned individuals. For dark skinned, we had 69% accuracy and light skinned was 55% accuracy. This was before we corrected for bias, and after correcting for bias we saw an decrease surprisingly in accuracy of 52% for dark skin, and 70% for light skin. We believe the reduction in the count of pictures in dark skin to keep the ratio equal between all pictures is the culprit to the low accuracy. To rectify this, we would need more data which was limited in this scenario, as well as perhaps tweak the hyperparameters to create a better network. This is proven by the increase in accuracy for lighter skin data, which increased from 55% to 70%.

References

[1] H. in the Loop, “Medical mask dataset | Humans in the Loop,” May 28, 2020. https://humansintheloop.org/resources/datasets/medical-mask-dataset/ (accessed Jun. 06, 2022).

[2] M. Vrigkas, E.-A. Kourfalidou, M. E. Plissiti, and C. Nikou, “FaceMask: A New Image Dataset for the Automated Identification of People Wearing Masks in the Wild,” *Sensors*, vol. 22, no. 3, p. 896, Jan. 2022, doi: 10.3390/s22030896. **https://mvrigkas.github.io/FaceMaskDataset/** (accessed Jun. 06, 2022).

[3] “Epoch vs Batch Size vs Iterations | by SAGAR SHARMA | Towards Data Science.” https://towardsdatascience.com/epoch-vs-iterations-vs-batch-size-4dfb9c7ce9c9 (accessed Jun. 06, 2022).

[4] COMP472 Artificial Intelligence (Summer 2022)
Lab Exercise #07: Introduction to Deep Learning