UNIVERSITAT DE BARCELONA
Facultat de Matemàtiques i Informàtica

Automatic Age Perception

Task 1

Gerard Castro Castillo [*] gcastrca25@alumnes.ub.edu
Àlex Pujol Vidal[**] apujolvi43@alumnes.ub.edu

Facultat de Matemàtiques i Informàtica
Universitat de Barcelona

March 24, 2023

1. Summary of contributions

In this report we briefly describe the problem of Automatic Age Perception and how we approach the issue of improving accuracy while minimizing the different bias scores of the given baseline model. The dataset we are dealing with is the Appa-Real Age Dataset. It consists on RGB images of 224x224 pixels, hence every image consist of a tensor of size 224x224x3 whose values are in a range of [0, 255]. For each image, some metadata is provided. We have three different categories:

- Gender: male or female.
- Ethnicity: Asian, Afro-American or Caucasian.
- Facial expression: neutral, slightly happy, happy or other.

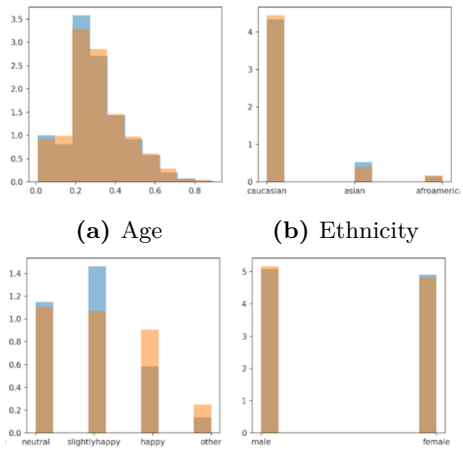
The target variable is age. We are expected to predict this value which is in the range [0, 1], that is real age normalized by a factor 100. The baseline model uses ResNet50 pre-trained to recognize faces and a naive data augmentation approach for the images for older people. To measure the performance of the model we rely on several bias metrics that compute the accuracy and the bias on every category.

Our model aims to improve the metrics by following the next scheme:

1. In order to address the problem of bias mitigation we studied the distribution of the sample images for each of the categories and applied a smart data augmentation strategy. As can be easily seen if figure , it is a very unbalanced dataset on different elements of the categories.

   We built a custom categorization system to upsample images from infra-represented compound groups, and possibly downsampling images from supra-represented compound groups.

2. The baseline model is based on the ResNet50 model. Many experiments were done by changing some of its hyperparameters, such as, learning rate and optimizer. Also, other backbone structure were implemented and we proposed a blending of the ResNet50 with a VGG model.



(a) Age **(b)** Ethnicity

(c) Facial expresion **(d)** Gender

Figure 1. Histograms of the original dataset for each category. In blue the training set, in orange the validation set.

[*] https://gcastro-98.github.io/
[**] https://socalesc.github.io/

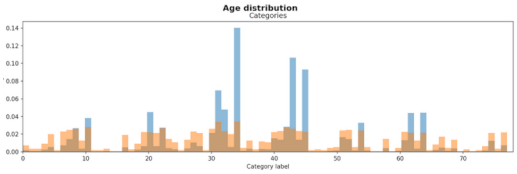


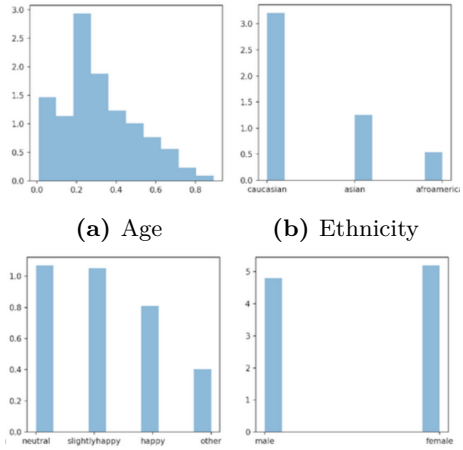
Figure 2. Comparison of compound categorical distributions before and after the DA.

1.1. Data augmentation strategy

During data exploration, compound categories were built in order to assess the distribution of the images over more complex groups. Images were divided into four different groups, from 0 to every 20 years, [0, 20), [20, 40), [40, 60) and [60, -). In total there are 4x2x3x4=96 compound groups, but only 78 were non-empty. The data augmentation (DA) strategy consisted not only augment the amount of samples of the most infra-represented compound categories, but also we experimented on dropping some samples in the most represented compound groups.

Six different image augmentation techniques were used: horizontal flip, change of brightness, Gaussian blur, random translation, random rotation and gray noise. An upper bound was defined as a hyperparameter. For instance, if the upper bound is $U = 200$, then if the number of images in the compound category is more than U, no augmentation is required and maybe some samples are dropped, otherwise each image of the group goes through the six-step augmentation. The amount of images generated from an image grows exponentially applying every possible combination of transformations on it (only rotation and translation is not applied at the same time, to avoid missing too much information). Between steps, the amount of images per compound group is rechecked and if U is surpassed the process for this group stops and moves to another compound group. A group may end with more than U images, in such case random dropping can be done. Notice that if necessary, the dropping is done at the end and random in order to avoid favoring any of the augmentation techniques already done. Figure shows the difference in the distribution of the images over the different compound categories.

By this approach we ensure not only more homogenization over complex combinations of categories, but each category, separately, becomes more balanced. Check figure to visualize the comparison of the histogram of the categories, before and after the DA:



(a) Age **(b)** Ethnicity

(c) Facial expresion **(d)** Gender

Figure 3. Histograms after data augmentation for each category.

1.2. Training strategy

Most of the effort relayed on the data augmentation and experimentation process. The training strategy remained the same as the baseline model. Since we are using a pretrained model with added layer at the end, it is divided in two stages. On the first stage of training the parameters of the pretrained model are fixed. Then when fitting, only the parameters of the extra layers are optimized. On the second stage, the whole model is able to learn from the data and all the parameters are tuned at the same time. The diagram visualizes this process.

In order to find the best parameters we followed an experiment based reasoning. First, we parted from the baseline ResNet50 model and modified some hyperparameters. Concretely we experimented with patience, number of epochs, learning rate and the optimizer. The thinking process for modifying each parameter is explained in the next section, as we were modifying each parameter according to the results of previous experiments. Once we found a proper set of hyperparameters, more experiments were done. This time we seek for a deeper modification of the model by changing the backbone of the first stage by using another pretrained model, that is a VGG19 instead of the ResNet50, and the second stage model by adding more layers.

2. Experiments and results

The experiments were done chronologically and we modified the parameters at the same pace of the con-

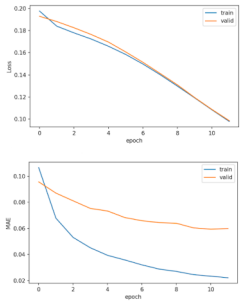


Figure 4. Training history for experiment 4.

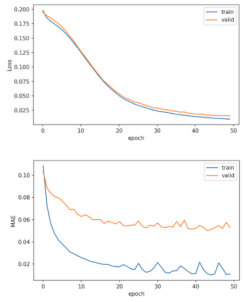


Figure 5. Training history for experiment 5.

clusions extracted from the experiments.

- **Experiment 1.** Consist on the baseline ResNet50 model without any data augmentation. The number of epochs is set at 12, the patience level for the early stopping at 5, the learning rate at 1e-5, and for the second stage training, three FC layers of sizes 518, 128 and 32, are appended to the ResNet50.

- **Experiment 2.** The naive data augmentation given at the starting-kit was implemented. It only augmented the images of older people. The MAE and the Bias improved but just a little. Hence, we designed and tried different combinations of data augmentation in our next experiments.

- **Experiment 3.** A first trial of our data augmentation improved by far the previous results. We set the upper bound of $U = 300$ samples per compound category and up to 12 transformations of an original image and not dropping overrepresented categories. That gave a total of 14571 samples.

- **Experiment 4** We tried adding more images by changing U to 400. However, worse results to the previous experiment were obtained. For efficiency and time we kept the previous DA of 300 samples and change, for the next experiment, the number of epochs and patience for the early stopping. Actually, a quick look at the train history curve of the second stage , suggested that we could increase the number of epochs and patience rather than the number of images.

- **Experiment 5.** Before changing the backbone, we want to find out what is the suitable amount of epochs, and also whether the learning rate can be sped up or improved. In this experiment we play with learning rate and further increase the number of epochs. The figure has a better shape than the previous one.

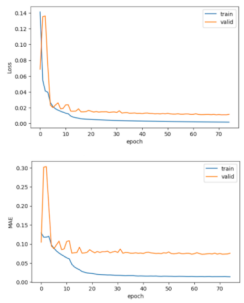


Figure 6. Training history for experiment 6.

- **Experiment 6.** As we can see in the previous experiment, we obtained a lower MAE, however some biases increased. It suggested us to augment more images setting $U = 350$, which gives a total of 16629 samples. Also, we increased the number of epochs but kept the patience level the same and played with the learning rate. Now we part from a higher learning rate of 5e-4 but added a ReduceLROnPlateau scheduler that progressively reduces the learning rate until 1e-5. As we can see, the experiment didn't work as expected. The scheduler does not seem to be a good idea. It is true that there are more samples now, but it is not likely to be the cause of the bad performance, since it was far worse than experiment 4.

- **Experiment 7.** Experiment 7 was launched simultaneously with experiment 6, so it inherited the bad performance from the previous one. We kept the DA process and the same hyperparameters as before, but we changed the backbone of the second stage of the model. In particular, we imported the trained Resnet50, froze its layers and added the same FC layers as the baseline

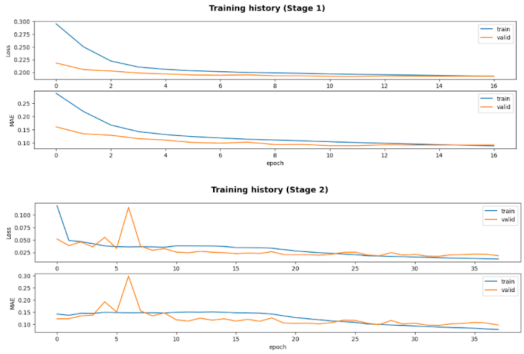


Figure 7. Training history for experiment 7. Stages 1 and 2.

model. Then, we added an extra 256-neurons dense FC layer in between the 512 and 128, and another Dropout layer afterward. We can also see a bad result, we also suspect that it can be an error in the implementation of the extra layer.

- **Experiment 8.** Now we experiment with the other part of the backbone, changing the pretrained model. This time we tried a VGG19 model instead of the ResNet50. The notebook of the experiment is attached to this document. However we run out of time to see the final results.

The table summarizes the results obtained in the experiments.

Table 1. Biases and error of the experiments.

Exp.	Ba	Bg	Be	Bf	MAE
1	6.408	0.499	1.438	0.471	9.849
2	2.877	0.600	1.149	0.797	9.056
3	1.814	0.575	0.428	0.280	5.527
4	3.849	0.094	0.798	0.372	5.991
5	3.179	0.038	0.459	0.765	5.285
6	4.567	0.433	0.593	0.254	9.060
7	3.797	0.489	1.755	0.403	9.712

3. Final remarks

After the experiments, we can extract the following conclusions. Our data augmentation approach was the most useful technique. Experiment 3 reach the best MAE and maybe with more epochs could farther improve. Remains to see the results of experiment 8 with the VGG, which we think they can be promising. Also, we propose as a further step a blending technique, that uses the model of experiment 3 and 8, which we suspect it can also improve the results.

The following kaggle notebook contains the experiment 3: `https://www.kaggle.com/gerardcastro/cv-task-1-experiment-3`