Practical Session Progress Discussion

Julio C. S. Jacques Junior

18/03/2022

Codalab (T.1)

- 22 participants
- 171 submissions

Results

#	User	Entries	Date of Last Entry	Ranking ▲	Gender (bias) ▲	Expression (bias) ▲	Ethnicity (bias) ▲	Age (bias) ▲	MAE ▲
1	adegaray	7	03/11/22	2.600000	0.131069 (3)	0.222518 (6)	0.038951 (1)	2.070795 (2)	4.299231 (1)
2	QJ	8	03/12/22	6.200000	0.044955 (1)	0.162736 (3)	0.539348 (13)	3.056905 (5)	6.340045 (9)
3	arturxe	1	03/10/22	6.600000	0.520513 (16)	0.198511 (4)	0.263101 (9)	1.803695 (1)	5.763641 (3)
4	Marcos	13	03/10/22	6.600000	0.178582 (5)	0.099530 (1)	0.330383 (10)	4.313029 (12)	5.947065 (5)
5	johnnynunez	40	03/13/22	7.600000	0.210905 (6)	0.350365 (12)	0.116024 (4)	3.700477 (10)	6.103415 (6)
6	arseniyy123	20	03/12/22	8.000000	0.105597 (2)	0.259403 (9)	1.198257 (19)	3.414286 (8)	5.458959 (2)
7	pere_luis	6	03/11/22	8.800000	0.294312 (10)	0.481102 (17)	0.097265 (3)	2.522612 (3)	6.479792 (11)
8	AlvaroLC	3	03/13/22	9.600000	0.714722 (18)	0.140125 (2)	0.654923 (15)	3.225240 (6)	6.220692 (7)
9	inaki_erregue	6	03/10/22	9.600000	0.853547 (20)	0.266028 (10)	0.228667 (7)	3.388354 (7)	5.785490 (4)
10	jha138	7	03/12/22	10.000000	0.411101 (14)	0.206496 (5)	0.082441 (2)	5.907411 (17)	6.663322 (12)
11	Iker	14	03/10/22	10.200000	0.243386 (9)	0.357067 (13)	0.203853 (6)	5.209423 (13)	6.340726 (10)
12	Volokin	7	03/13/22	10.800000	0.508896 (15)	0.403236 (15)	0.462716 (12)	2.663317 (4)	6.319689 (8)
13	ammtomi	4	03/13/22	11.600000	0.242992 (8)	0.722199 (19)	0.259435 (8)	3.543867 (9)	7.070488 (14)
14	mdlt	5	03/13/22	11.600000	0.139774 (4)	0.421380 (16)	0.622958 (14)	3.763480 (11)	6.779570 (13)
15	xavidejuan	5	03/09/22	11.600000	0.323289 (12)	0.230054 (7)	0.141708 (5)	5.466550 (16)	7.466782 (18)
16	lorenzovigo	1	03/08/22	12.800000	0.319109 (11)	0.253672 (8)	0.358137 (11)	5.448703 (15)	9.240306 (19)
17	mertmecit	1	03/13/22	15.000000	0.235272 (7)	0.658474 (18)	0.777892 (16)	6.300268 (18)	7.241075 (16)
18	e1307685	12	03/13/22	15.200000	0.781770 (19)	0.280844 (11)	0.890152 (17)	5.299491 (14)	7.149036 (15)
19	rudiboi	6	03/10/22	16.400000	0.380944 (13)	0.384031 (14)	0.940844 (18)	9.023186 (20)	7.451245 (17)
20	juliojj	1	02/28/22	19.800000	0.628002 (17)	0.819692 (20)	2.447512 (22)	8.988890 (19)	11.141265 (21)
21	aa.dudek5	2	03/10/22	20.600000	2.000426 (21)	1.568660 (21)	1.806945 (20)	13.798619 (21)	10.791063 (20)
22	noahjadallah	2	03/10/22	21.800000	4.417457 (22)	2.949319 (22)	2.223809 (21)	19.004131 (22)	14.001426 (22)

Your feedback is important

- Did you find **the problem** interesting?
- Is there anything that could be **improved/changed** in this process?

General notes

- Quick feedback w.r.t Task 1
- Take the notes into account when delivering Task 2 (and the optional task)

General notes

- The **code** should complement the **report**
  - The report document should contain **all the details** required to understand the proposed solution and results.
    - Do not assume I know the model you use “we include a new dropout layer just after the last FC layer”
      - What model?
    - Do not assume I know the training strategy: “we applied training strategy 2”

Imagine you are reporting a procedure that should be **reproducible from the provided information**

General notes

- What most of you did that is not a standard in research
  - Find the best model / hyperparameters base on the evaluation performed on the **Test set.**
  - The search of the best model / hyperparameters should **use the train/validation set**.
  - Test set should be used only after the model is trained and hyperparameters defined.

- Perform general and high level comments like:
  - Model X obtained overall lower bias score and MAE
    - **Be curious!**
    - *What could explain that?*
    - *Did you solution addressed the problem well?*
    - *Could it be related with the fact that you augmented category 1 only?*
    - You can go be beyond simple and general explanations.

General notes

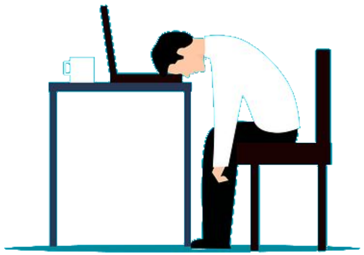
- Things you could have done before start playing with data augmentation
  - Compare different **backbones (VGG vs. Resnet)**
  - Compare the same model with/without **transfer learning**
  - Compare the same model with **transfer learning from different datasets (imagenet, faces)**
  - Fix the backbone, include **new layers or regularizers** and compare (results, time, etc)
  - Different **losses**: MSE vs MAE
  - Different **optimizers** (Adam vs SGD)
  - Computational **time** vs. number of **parameters** vs. **result**

Available models

Model	Size (MB)	Top-1 Accuracy	Top-5 Accuracy	Parameters	Depth	Time (ms) per inference step (CPU)	Time (ms) per inference step (GPU)
Xception	88	79.0%	94.5%	22.9M	81	109.4	8.1
VGG16	528	71.3%	90.1%	138.4M	16	69.5	4.2
VGG19	549	71.3%	90.0%	143.7M	19	84.8	4.4
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
ResNet50V2	98	76.0%	93.0%	25.6M	103	45.6	4.4
ResNet101	171	76.4%	92.8%	44.7M	209	89.6	5.2
ResNet101V2	171	77.2%	93.8%	44.7M	205	72.7	5.4
ResNet152	232	76.6%	93.1%	60.4M	311	127.4	6.5
ResNet152V2	232	78.0%	94.2%	60.4M	307	107.5	6.6
InceptionV3	92	77.9%	93.7%	23.9M	189	42.2	6.9
InceptionResNetV2	215	80.3%	95.3%	55.9M	449	130.2	10.0
MobileNet	16	70.4%	89.5%	4.3M	55	22.6	3.4
MobileNetV2	14	71.3%	90.1%	3.5M	105	25.9	3.8
DenseNet121	33	75.0%	92.3%	8.1M	242	77.1	5.4
DenseNet169	57	76.2%	93.2%	14.3M	338	96.4	6.3
DenseNet201	80	77.3%	93.6%	20.2M	402	127.2	6.7
NASNetMobile	23	74.4%	91.9%	5.3M	389	27.0	6.7
NASNetLarge	343	82.5%	96.0%	88.9M	533	344.5	20.0
EfficientNetB0	29	77.1%	93.3%	5.3M	132	46.0	4.9
EfficientNetB1	31	79.1%	94.4%	7.9M	186	60.2	5.6
EfficientNetB2	36	80.1%	94.9%	9.2M	186	80.8	6.5
EfficientNetB3	48	81.6%	95.7%	12.3M	210	140.0	8.8
EfficientNetB4	75	82.9%	96.4%	19.5M	258	308.3	15.1
EfficientNetB5	118	83.6%	96.7%	30.6M	312	579.2	25.3
EfficientNetB6	166	84.0%	96.8%	43.3M	360	958.1	40.4
EfficientNetB7	256	84.3%	97.0%	66.7M	438	1578.9	61.6
EfficientNetV2B0	29	0.787	0.943	7,200,312	-	-	-
EfficientNetV2B1	34	0.798	0.950	8,212,124	-	-	-
EfficientNetV2B2	42	0.805	0.951	10,178,374	-	-	-
EfficientNetV2B3	59	0.820	0.958	14,467,622	-	-	-
EfficientNetV2S	88	0.839	0.967	21,612,360	-	-	-
EfficientNetV2M	220	0.853	0.974	54,431,388	-	-	-
EfficientNetV2L	479	0.857	0.975	119,027,848	-	-	-
EfficientNetV2B0	29	0.787	0.943	7,200,312	-	-	-
EfficientNetV2B1	34	0.798	0.950	8,212,124	-	-	-
EfficientNetV2B2	42	0.805	0.951	10,178,374	-	-	-
EfficientNetV2B3	59	0.820	0.958	14,467,622	-	-	-
EfficientNetV2S	88	0.839	0.967	21,612,360	-	-	-
EfficientNetV2M	220	0.853	0.974	54,431,388	-	-	-
EfficientNetV2L	479	0.857	0.975	119,027,848	-	-	-

General notes

- Totally **avoid** <u>using the starting kit as it is</u>, and just change some parameters
  - **Same model** architecture



```python
# Using the FC layer before the 'classifier_low_dim' layer as feature vector
fc_512 = model.get_layer('dim_proj').output

# adding a dropout layer to minimize overfiting problems
dp_layer = Dropout(0.5)(fc_512)

# adding a few hidden FC layers to learn hidden representations
fc_128 = Dense(128, activation='relu', name='f_128')(fc_512)
fc_32 = Dense(32, activation='relu', name='f_32')(fc_128)

# Includint an additional FC layer with sigmoid activation, used to regress
# the apparent age
output = Dense(1, activation='sigmoid', name='predict')(fc_32)
```

  - **Just changing the hyperparameters** (learning rate, batch size, num of epochs, etc)
  - Applying **exactly the same training strategy** (e.g., *new layers* + *half of the network*)
  - **Just changing the parameters of data transformation** (e.g., filter size of Gaussian blur)

```python
X_train_augmented.append(cv2.GaussianBlur(X_train[i], (5,5),1.0))
X_train_augmented.append(cv2.GaussianBlur(X_train[i], (7,7),1.0))
```

Page limits

- Please, do not exceed the page limit
  - Unfair with those who delivered a 4 pages report
  - Requires more time to evaluate
    - 12 groups

Help the reader (proposed solution and experiments)

- Main points to be detailed in the report before discussing the results
  - **Model** architecture (backbone? Adaptation?)
  - **Training** strategy (1 or multiple stages?)
    - **Hyperparameters**, optimizer and loss?
  - Data **augmentation**
    - **Transformations** (flip, rotation, etc) ← “standard, but not in the case of this exercise”
    - **Categories** (age, gender, etc)
    - Defined **strategy** (randomly, based on data distribution, etc)
  - Custom loss (*Task 2*)
  - Data augmentation **&** Custom loss (*optional exercise*), if this will be the case

Optimize your report → avoid redundancy (you have 4pg)

Ranking	Gender bias	Expression bias	Ethnicity bias	Age bias	MAE
4.4	0.22	0.43	0.39	3.86	10.5

Ranking	Gender bias	Expression bias	Ethnicity bias	Age bias	MAE
4.4	0.32	0.25	0.36	5.45	9.24

Ranking	Gender bias	Expression bias	Ethnicity bias	Age bias	MAE
3.8	0.32	0.23	0.14	5.47	7.47

Model	*Data Aug.*	*Gender bias*	*Expression bias*	*Ethnicity bias*	*Age bias*	*MAE*
Baseline	Age	**0.22**	0.43	0.39	3.86	10.5
Tuned	Age					
Baseline	Complete	0.32	0.25	0.36	**5.45**	9.24
Tuned	Complete	0.32	**0.23**	**0.14**	5.47	**7.47**

Help the reader (e.g., highlight best results)

Model	Age bias	Gender bias	Ethnicity bias	Expression bias	MAE
1	9.51	1.33	3.25	1.86	20.09
2	6.43	0.16	1.53	0.34	12.31
3	7.47	0.16	0.76	1.09	9.42
4	2.52	0.29	0.09	0.48	6.47

Which option help us to identify the "best" solution easier?

Model	Age bias	Gender bias	Ethnicity bias	Expression bias	MAE
1	9.51	1.33	3.25	1.86	20.09
2	6.43	0.16	1.53	0.34	12.31
3	7.47	0.16	0.76	1.09	9.42
4	2.52	0.29	0.09	0.48	6.47

Help the reader (clearly define/present the experiments)

Model	Age bias	Gender bias	Ethnicity bias	Expression bias	MAE
1	9.51	1.33	3.25	1.86	20.09
2	6.43	0.16	1.53	0.34	12.31
3	7.47	0.16	0.76	1.09	9.42
4	2.52	0.29	0.09	0.48	6.47

Model	Learning rate	Data augmentation	Regularizers	Dropout	Train all layers
1	1e-5	No	No	No	No
2	1e-5	Yes	No	No	No
3	1e-4	Yes	Yes	Yes	No
4	1e-5	Yes	Yes	Yes	Yes

Additional information can help and support the analysis.

Help the reader (clearly define/present the experiments)

Model	Learning rate (initial)	Batch size	Epochs	# images	Gender bias	Expression bias	Ethnicity bias	Age bias	MAE
A_s1	3E-05	16	50	9560	0.55	3.11	3.68	13.74	16.12
A_s2	1E-05	16	30	10710	0.09	0.58	1.01	5.73	8.24
A_s3	1E-05	16	3	10710	0.41	0.60	1.7	4.46	7.84
B_s1	3E-05	16	50	9560	0.12	1.77	1.45	8.46	11.68
B_s2	1E-05	12	50	8590	0.17	0.25	1.10	6.52	8.97
B_s3	1E-05	12	21	6885	0.41	0.63	0.88	5.19	6.77
C_s1	3E-05	16	50	9560	1.17	2.64	2.41	14.74	15.51
C_s2	1E-05	12	50	8590	0.30	0.75	0.95	6.04	8.40
C_s3	**1E-05**	**12**	**50**	**6885**	**0.18**	**0.10**	**0.33**	**4.31**	**5.95**

All together + extensive set of experiments

Help the reader (clearly define/present the experiments)

Data augm.	*Learning rate*	*Batch size*	*Loss function*	*Gender bias*	*Expression bias*	*Ethnicity bias*	*Age bias*	*MAE*
I	*1e-5*	*32 - 16*	MSE	0.994277	0.401712	0.140987	2.711542	6.257771
I	*1e-5*	*64 - 16*	MSE	0.438000	0.537266	1.082621	2.969890	6.348123
I	*1e-5*	*64 - 32*	MSE	0.359906	0.461378	0.848432	6.928862	7.937758
I	*1e-4*	*64 - 32*	MSE	0.246278	0.434221	0.870553	7.132548	7.913878
II	*1e-5*	*32 - 16*	MSE	0.508897	0.403237	**0.462716**	**2.663317**	6.319689
II	*1e-5*	*32 - 16*	MAE	0.714722	**0.140125**	0.654923	3.225239	6.220692
II	*1e-5*	*64 - 32*	MSE	**0.067334**	0.322087	0.606145	5.052895	7.252613
II	*1e-4*	*32 - 16*	MSE	0.686798	0.362448	0.566761	3.598495	**6.071322**
II	*1e-4*	*32 - 16*	MAE	0.681565	0.653679	0.996174	3.846475	6.805394

In your opinion, what is the best model?

All together + extensive set of experiments

Help the reader (clearly define/present the experiments)

Bias	Pre-trained	A	B	C
Age bias (Ba)	8,98889	7,34068	5,69922	4,44693
Gender bias (Bg)	0,62800	1,12575	0,12704	0,13599
Ethnicity bias (Be)	2,44751	1,73011	0,33787	0,67792
Face Expression bias (Bf)	0,81969	0,43297	0,66389	1,14408

Best vs. **Worst** results

Help the reader (clearly define/present the experiments)

Model	*Learning rate*	*Training strategy*	*Gender bias*	*Expression bias*	*Ethnicity bias*	*Age bias*	*MAE*
X	*1e-5*	*2*	*0.509502*	*1.384053*	1.063591	8.491639	7.507680
Y	*1e-5*	*2*	**0.105597**	**0.259403**	1.198257	**3.414286**	5.458959
Z	*1e-4*	*2*	0.293883	0.436619	0.855126	4.192811	6.455369
L	*1e-5*	*2*	0.590356	0.446444	**0.692233**	6.787660	**5.285532**

Additional information **could** support the analysis.

Model	Learning rate	Data augmentation	Regularizers	Dropout	Train all layers
1	1e-5	No	No	No	No
2	1e-5	Yes	No	No	No
3	1e-4	Yes	Yes	Yes	No
4	1e-5	Yes	Yes	Yes	Yes

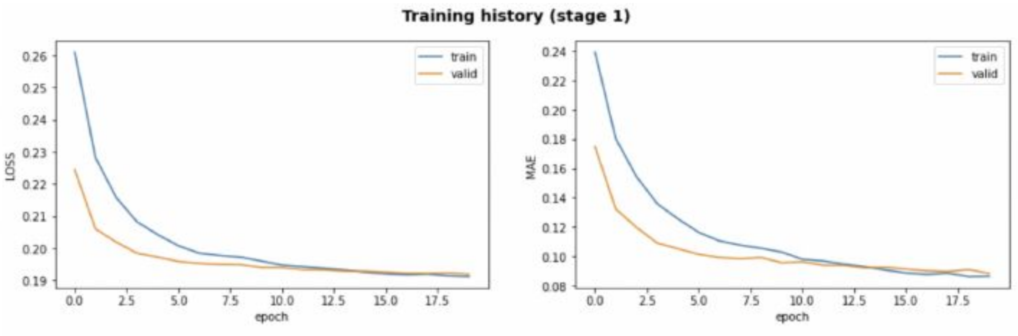
Help the reader (clearly define/present the experiments)

Method	MAE	Age bias	Gender Bias	Ethnicity Bias	Face expression bias
Connected methods small dataset: flip, blur and translation	16.40483976	4.374016125996907	0.35525894	0.8886521657307943	0.664334774017334

Method	MAE	Age bias	Gender Bias	Ethnicity Bias	Face expression bias
Basic whole dataset model (4k observations) with no data augmentation	13.77970156	4.298641204833984	0.1668024	0.5610771179199219	1.2939891815185547
Totally augumented dataset (4k)	**13.54146676**	6.9638926188151045	0.42714214	0.5652459462483724	0.6126677195231119
"Half-augmented" model (2k real photos+ 2k augmented)	13.59614196	6.295828501383464	0.5167999	1.237823486328125	1.2830932935078938
Basik 2k model with no augumented data	15.35284666	5.085680643717448	0.47130203	1.5034319559733074	1.035815715789795

Low resolution in the report

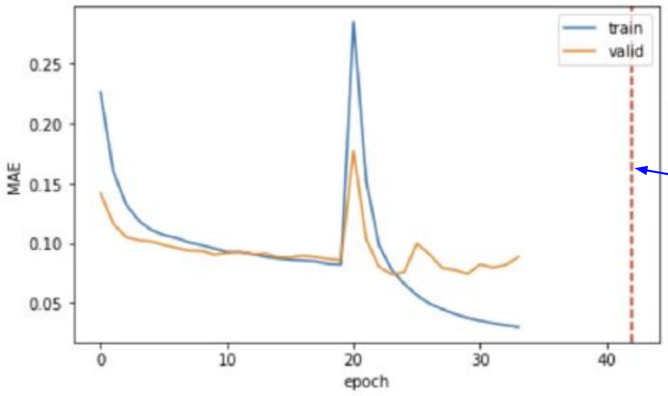
Help the reader (clearly define/present the experiments)



Do not show an image (or table) if you are not going to **discuss it in the text**!

"As it can be seen in Figure x…"

Help the reader (clearly define/present the experiments)



Avoid **distracting** representations that are not discussed

“The dashed line in Figure x represents…”

Simple yet effective solutions

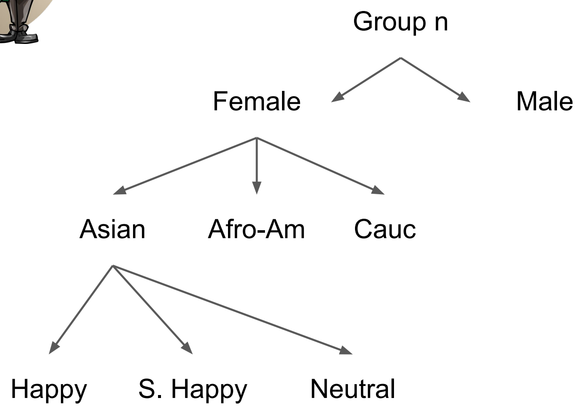
Model	1st Stage	2nd Stage	Gender bias	Expression bias	Ethnicity bias	Age bias	MAE
M1	No DA	No DA	0.6280	0.8197	2.4475	8.9889	11.1413
M2	No DA	Small DA	0.5132	0.8045	1.3245	8.0390	9.8163
M3	Full DA	Full DA	0.3442	0.3673	0.4700	2.2580	**4.1436**
M4	Full DA	No DA	**0.1310**	0.2225	**0.0389**	**2.0707**	4.2992
M5	Full DA	No DA	0.2550	**0.0923**	1.0313	2.0837	4.3221

Data augmentation when having less parameters to train
(backbone is frozen)

Go deep to provide better explanations

	Model 1	Model 2	Model 3	Model 4
Age Bias	**9.51**	**6.43**	**7.47**	**2.52**
Group 1 MAE	14.04	14.05	12.40	7.08
Group 2 MAE	19.18	9.24	5.94	5.29
Group 3 MAE	23.00	14.54	10.97	7.39
Group 4 MAE	31.80	21.95	20.41	10.23
Gender Bias	**1.33**	**0.16**	**0.16**	**0.29**
Female MAE	20.73	12.23	9.33	6.62
Male MAE	19.40	12.39	9.50	6.32
Ethnicity Bias	**3.25**	**1.53**	**0.76**	**0.09**
Asian MAE	15.65	10.21	8.36	6.52
Afroamerican MAE	15.81	10.52	8.90	6.38
Caucasian MAE	20.54	12.51	9.51	6.47
Expression Bias	**1.86**	**0.34**	**1.09**	**0.48**
Happy MAE	20.84	12.56	9.56	6.67
S. Happy MAE	20.88	12.11	9.17	6.66
Neutral MAE	19.38	12.18	9.16	6.21
Other MAE	17.65	12.66	11.23	5.96

Should you go **deeper**?



It is up to you. But **why not?**

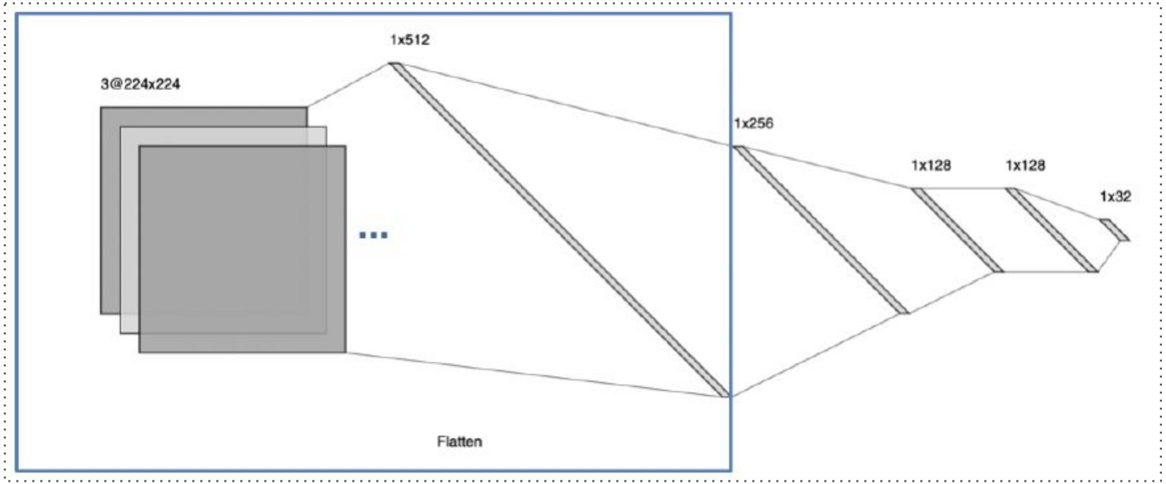
Important parameters need to be discussed

Strategy	Age	Factor
By age intervals	*(40,60)*	*0.8*
By age metadata	***Metadata***	***Factor***
	Other	*0.8*
	Asian	*0.7*
	Afroamerican	*1*
By Subsets	***Age Interval***	***Factor***
Asian	*(0,20)*	*0.9*
Asian	*(40,100)*	*0.9*
Afroamerican	*(0,100)*	*1*
Caucasian	*(10,20)*	*0.2*
Caucasian	*(60,100)*	*0.2*

Ex:

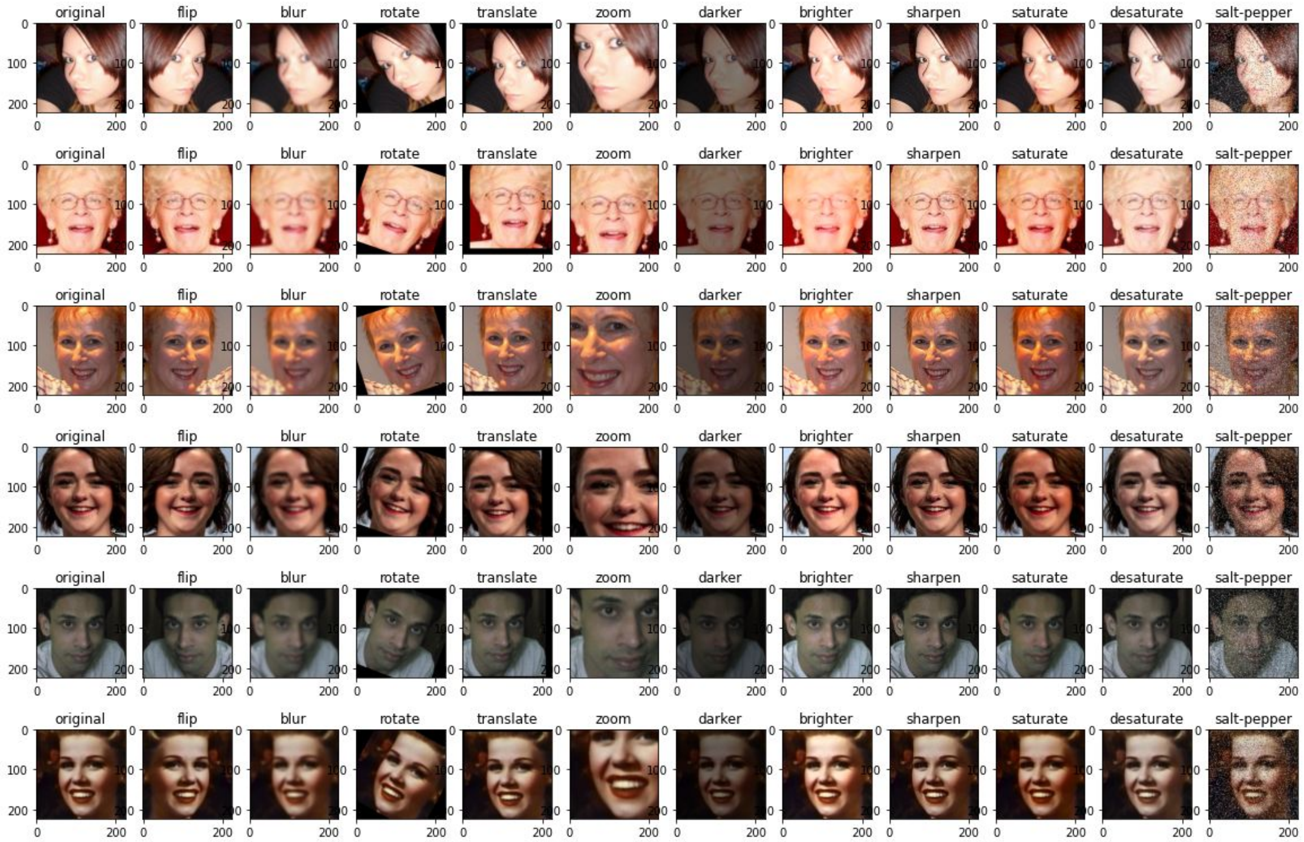
- parameters were empirically defined based on data distribution;

Graphical representations are more than welcome



Graphical illustration of the proposed model

Graphical representations are more than welcome



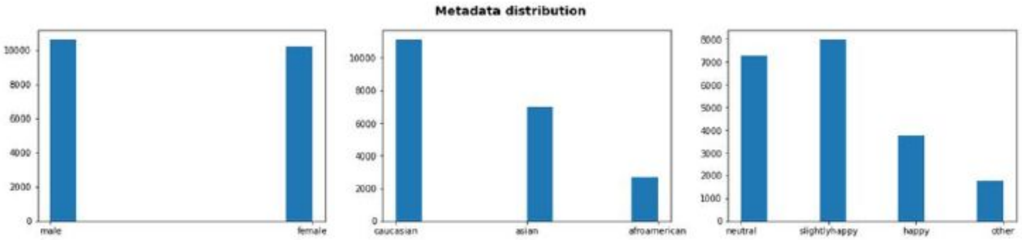
In your opinion, what else could be done here to increase variability?

Graphical representations are more than welcome

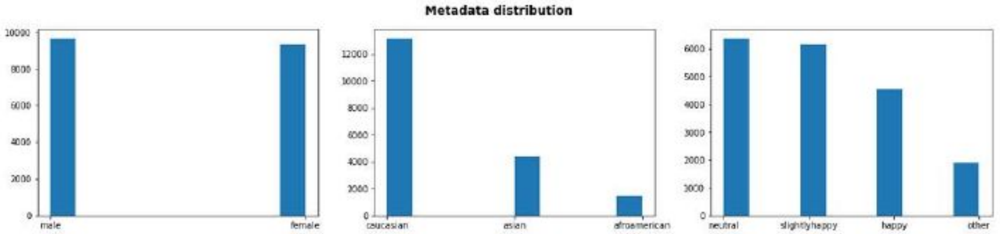
+ random (horizontal) flip
would increase variability

Graphical representations are more than welcome

Data augmentation 1:



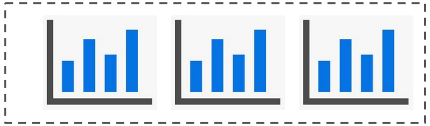
Data augmentation 2:



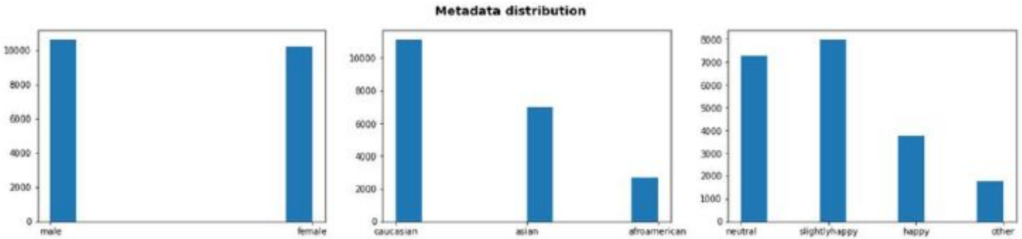
What do you “miss” here?

Graphical representations are more than welcome

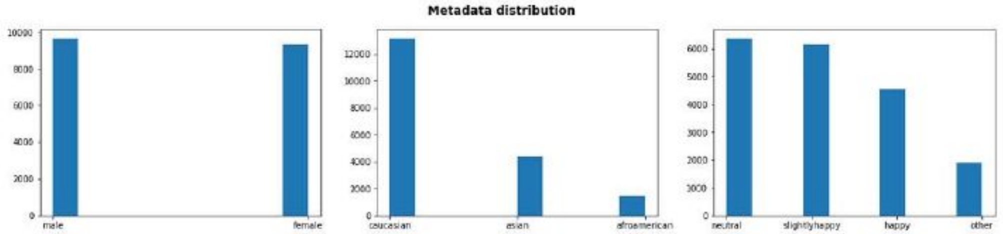
Original distribution →



Data augmentation 1:



Data augmentation 2:



You have limited space

```
mae: 0.0762 - val_loss: 0.1937 - val_mae: 0.0955
Epoch 6/50
255/255 [==============================] - 17s 68ms/step - loss: 0.1882
- mae: 0.0719 - val_loss: 0.1936 - val_mae: 0.0941
Epoch 7/50
255/255 [==============================] - 17s 68ms/step - loss: 0.1875
- mae: 0.0690 - val_loss: 0.1934 - val_mae: 0.0936
Epoch 8/50
255/255 [==============================] - 15s 60ms/step - loss: 0.1866
- mae: 0.0642 - val_loss: 0.1936 - val_mae: 0.0942
Epoch 9/50
255/255 [==============================] - 15s 60ms/step - loss: 0.1859
- mae: 0.0609 - val_loss: 0.1941 - val_mae: 0.0954
Epoch 10/50
255/255 [==============================] - 18s 69ms/step - loss: 0.1852
- mae: 0.0567 - val_loss: 0.1932 - val_mae: 0.0931
Epoch 11/50
255/255 [==============================] - 16s 61ms/step - loss: 0.1847
- mae: 0.0551 - val_loss: 0.1935 - val_mae: 0.0938
Epoch 12/50
255/255 [==============================] - 16s 61ms/step - loss: 0.1842
- mae: 0.0520 - val_loss: 0.1942 - val_mae: 0.0961
Epoch 13/50
255/255 [==============================] - 16s 61ms/step - loss: 0.1835
- mae: 0.0488 - val_loss: 0.1942 - val_mae: 0.0956
Epoch 14/50
255/255 [==============================] - 17s 66ms/step - loss: 0.1832
- mae: 0.0474 - val_loss: 0.1946 - val_mae: 0.0964
Epoch 15/50
255/255 [==============================] - 16s 61ms/step - loss: 0.1829
- mae: 0.0460 - val_loss: 0.1938 - val_mae: 0.0949
Epoch 15: early stopping
```

Avoid showing training history in the report document.

- Not attractive
- Use of space

Unexpected transformations for the problem at hand

original

vertical flip

wide angle rotation

Divide to conquer

Attribute-based analysis

	Baseline	Shearing & Cropping	Illumination Changes	Horizontal Flip	Blur	Translation	Rotation
Age Bias	8.99	7.20	**6.19**	8.13	7.84	8.50	7.10
Gender Bias	0.63	0.61	**0.33**	0.34	0.44	0.98	0.40
Ethnicity Bias	2.45	0.84	1.00	0.98	1.09	1.40	**0.74**
Face Expression Bias	0.82	0.39	0.41	0.34	**0.30**	0.41	0.33
Test MAE	11.14	10.33	**10.04**	10.36	10.33	14.10	10.47

Group-based analysis

	Age Analysis				Gender Analysis	
	Group 1	Group 2	Group 3	Group 4	Male	Female
Baseline	11.29	8.09	12.66	25.62	10.84	11.47
Shearing & Cropping	11.09	7.58	11.44	22.71	10.04	10.65
Illumination Changes	11.24	7.76	**10.50**	**20.08**	**9.84**	**10.25**
Horizontal Flip	**9.79**	7.65	12.47	23.02	10.19	10.53
Blur	11.84	**7.44**	11.03	22.86	10.12	10.56
Translation	12.39	11.53	16.78	27.07	13.63	14.60
Rotation	10.72	7.95	11.93	21.75	10.28	10.68

We **cannot** fine-tune our model using the **Test set**

	Baseline
Age Bias	8
Gender Bias	
Ethnicity Bias	
Face Expression Bias	
Test MAE	

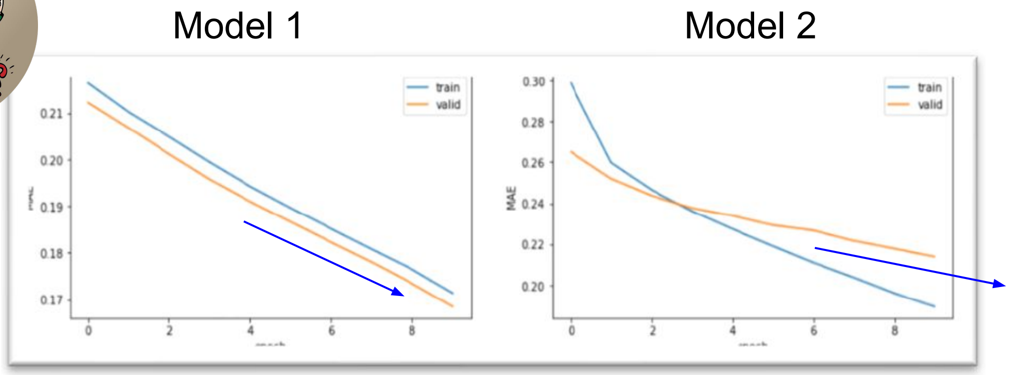
Basel						
Shearin						
Illuminatio						
Horizont						
Blur						
Translation	12.39	11.53	16.78	27.07	13.63	14.60
Rotation	10.72	7.95	11.93	21.75	10.28	10.68

Very interesting finding

- "We had a model with an age bias of **4.88** (almost half of the baseline) but a Test MAE of **17.76** (MAE baseline = 11.14) which showed us that **we can decrease the age bias by predicting the age equally bad for each subgroup**."

What could be **a possible limitation (or weak point)**, taking into account our average ranking metric?

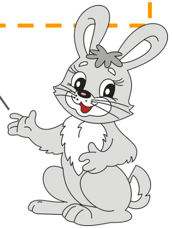
Be curious and investigate any possible source of problem



- Unusual behavior (left)
- Trained for very few epochs (inconclusive).
- Both models were still learning.

Revisit Task 2 & the optional exercise

Deadlines:
- **Task 2:** ~~Apr-1st~~ Apr-3rd
- **Optional:** Apr-10



- Task 2
  - **CUSTOM LOSS**
  - **WITHOUT** data augmentation
  - Which give you better results, data augmentation or custom loss?
    - Baseline (starting-kit) vs. Task 1 vs. Task 2

Do not wait for the last week to start playing with Task 2

- Optional exercise
  - Exploit your creativity as much as you can
  - Which give you better results, data augmentation or custom loss or all together?
    - Baseline (starting-kit) vs. Task 1 vs. Task 2 vs. "Task 3"