Eigenfaces – Age Group Classification

Introduction

Age estimation is a widely studied task related to facial recognition, with applications in domains such as biometrics, and human computer interaction. Estimating age from facial images suffers from many of same challenges as face recognition, such as variations in appearance caused by pose, lighting, and facial accessories (i.e. glasses) or facial hair; and has similar issues around the lack of diversity in training datasets.

Dataset Description

The dataset supplied is a compressed zip file containing 20,000+ colour face images, each has been cropped to size 200x200 and contains only face regions. Files are named as [age]_[gender]_[race]_[datestamp].jpg.

Data Preparation

We load all the images files together with its attributes, which are derived from the filename. To get a good distribution, all filenames as shuffled before commence loading. To expedite model training, all images are also resized to 32x32 upon loading and are converted to grayscale after being loaded.

Method Chosen

Age Group Classification

The primary response of the model is to determine the age of a person in an image. We would approach this as a classification task. However, instead of classifying each age as a class, a new class named 'decade' are created to group age into multiple of 10. For example, age 0 to 9 is class "0", age 10 to 19 is class "1", and so on.

Eigenfaces

As the dataset is relatively huge (20,000+), we decided to adopt "eigenfaces" method to capture the main features from the images. After the process, we derive that 66 components, or over 6.45% of the total dimensions explain 90% of the variation; 165 components (16.11%) to explain 95% and 529 components (51.66%) to explain 99%. This means 35% more data is needed to explain the extra 4% differences between 95% and 99%.

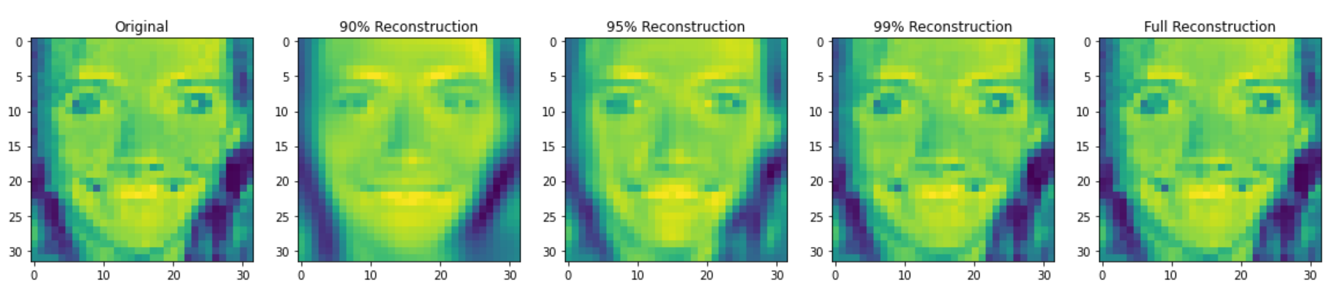

Figure 4.1: An example showing image reconstruction

CKNN Model

In consideration of the size of the dataset, and computational constraints, a CKNN classifier is selected as an approach for this age group classification task.

Hold-out Protocol

Splitting Data

Using a hold-out evaluation protocol, data are split into training (70%), validation (15%) and testing (15%) sets. We start of by training a CKNN Classifier (10 neighbours with inverse weighted distance) to test out different variance set we have generated (i.e. 90%, 95%, 99% and raw image data). We have also specified parameter "weights" with value "distance" to give the closer neighbour points a greater influence.

From these, the result from 95%-variance dataset is comparable to 99%-variance dataset with small difference. We proceed in using the 95%-variance dataset to try to search for an optimal value of number of neighbours parameter. The result is summarised as followed:

Model	Input Data	Misclassified / Total samples	Accuracy
CKNN, with n_neighbor=10, weights='distance'	90%	2188 / 3558	0.3850
CKNN, with n_neighbor=10, weights='distance'	95%	2210 / 3558	0.3789
CKNN, with n_neighbor=10, weights='distance'	99%	2197 / 3558	0.3825
CKNN, with n_neighbor=10, weights='distance'	FULL	2193 / 3558	0.3836
CKNN, with n_neighbor=5, weights='distance'	95%	2269 / 3558	0.3623
CKNN, with n_neighbor=20, weights='distance'	95%	2165 / 3558	0.3915

Figure 4.2: Validation Result of different models fitted

Cross-fold Protocol

Cross-fold design

Still fitting a model to determine the age of a given face, however this time we train the model progressively by training faces of 4 races and test the model on faces of the unseen race. This is done by creating folds with each fold containing faces of one race. The model is trained on 4 of the folds and test on the remaining folds. This is repeated until all folds are the test set in turn. This is to compare the effectiveness of the model in predicting out-of-sample data.

Again, CKNN classifier is fitted with number of neighbours=20 and using inverse weight is fitted here, on 95%-variance dataset, just to make the models comparable to each other when we perform evaluation.

Fold	Model	Input Data	Misclassified / Total samples	Accuracy
1	CKNN, with n_neighbor=50, weights='distance'	95%	7220 / 10079	0.2837
2	CKNN, with n_neighbor=50, weights='distance'	95%	3074 / 4536	0.3223
3	CKNN, with n_neighbor=50, weights='distance'	95%	1646 / 3434	0.5207
4	CKNN, with n_neighbor=50, weights='distance'	95%	2421 / 3975	0.3909
5	CKNN, with n_neighbor=50, weights='distance'	95%	890 / 1692	0.4740
			(weighted average)	0.5010

Figure 4.3: Test Result of Cross-Fold

Evaluation

Hold-out evaluation

The performance of the model fitted using hold-out evaluation protocol on test dataset is summarised below:

```
Total test samples: 3558
Misclassified samples: 2122
Test Accuracy: 0.4035975267003935
```

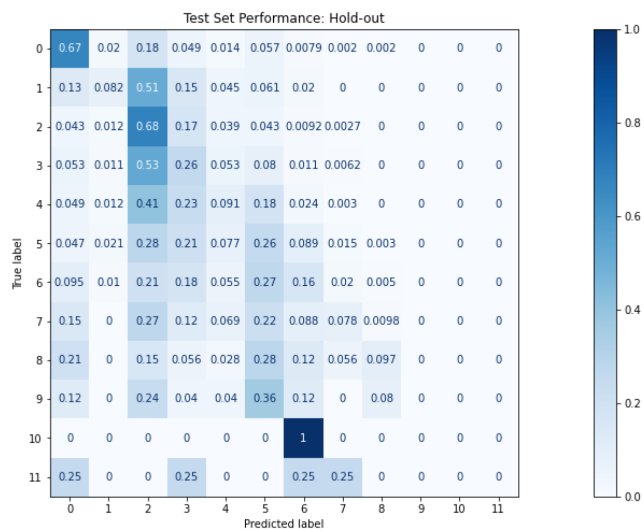

Figure 4.4: Test Result of Model (Hold-out)

Overall, the model is performing quite poorly in determining the age group of a given face image (<50%), especially faces of nonagenarians and centenarians. This may be due to the class imbalance in the dataset – low number of samples to train for older age group. Alternatively we may need to reclassifying the age range for grouping, or try out different classifier.

Cross-fold evaluation

One notable difference is that the performance is not consistent across all folds. For classification problem, stratified sampling is recommended to ensure folds created have the same percentage of samples of each class as the complete set. In addition, it is generally recommended to keep each fold the same size, which may have impacted on the performance of the model too, given that the model has different number of samples to train on.

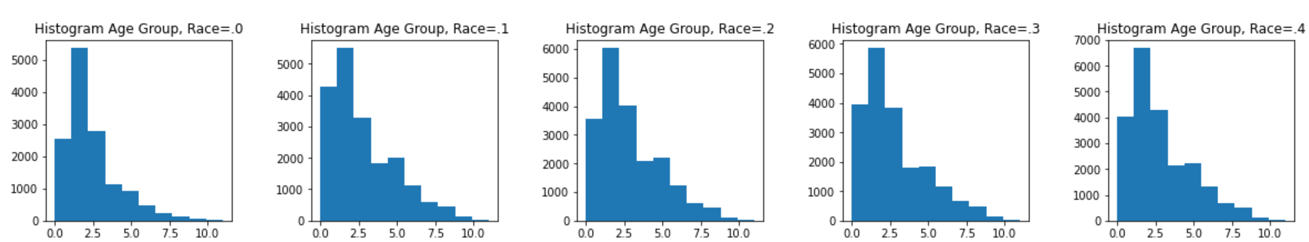

Figure 4.5: Distribution of Age Group by Race