Facial Recognition and Identification System

Project Report

Group 2

Prepared by:

Devanshi Shah – 301175169

Shrikant Kale - 301150258

Jefil Tasna John Mohan - 301149710

Hitesh Dharmadhikari – 301150694

Nestor Romero Leon – 301133331

Prepared for:

Professor Tenzin Jinpa

Unsupervised and Reinforcement Learning – Section 4

Centennial College

Fall Term – 2022

Contents

[1. Introduction 3](#_Toc121916939)

[2. Data Exploration 3](#_Toc121916940)

[3. Data Preprocessing 4](#_Toc121916941)

[3.1 Dimensionality Reduction 5](#_Toc121916942)

[4. Data Clustering 5](#_Toc121916943)

[4.1 K-Means Clustering 6](#_Toc121916944)

[4.2Agglomerative Hierarchical Clustering 7](#_Toc121916945)

[4.3 Gaussian Mixture Clustering 8](#_Toc121916946)

[5. Multiclass Classifier – ANN 9](#_Toc121916947)

[6. Model Results 11](#_Toc121916948)

[7. Conclusions 12](#_Toc121916949)

# 1. Introduction

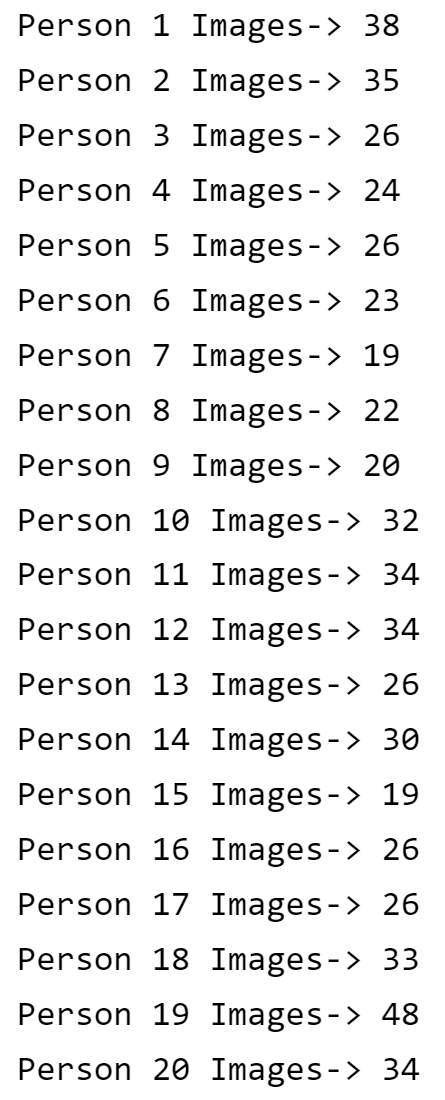
The objective of the project is to develop a facial recognition and identification system that uses the Unsupervised Learning techniques and models learned in class such as Principal Component Analysis, Dimensionality Reduction and Clustering.

For the project, the team received a dataset containing multiple images from a group of people to be used as the input for the model. This dataset needs to be processed and prepared to use it effectively in a machine learning model; as any other real-world data requires cleaning, standardization and reshaping as appropriate.

# 2. Data Exploration

The **umist\_cropped.mat** dataset contains the data that is used in the project to train the machine learning model. After loading the dataset, the main findings of the data can be summarized as:

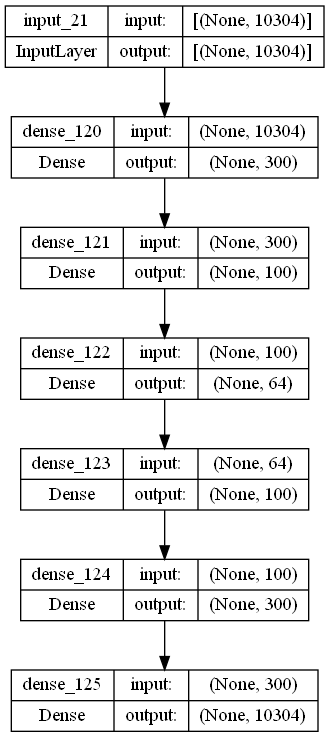
* The dataset contains a total of 575 images organized into 20 groups. All images in a particular group belong to the same person and each image contains a different profile of their face.
* The dataset is imbalanced, even though there are 20 groups of images per person, each group has a different size; the image count per group ranges between 19 and 48, requiring preprocessing before attempting to train a model.
* Image size is uniform in the dataset; each image has a resolution of 120 x 92 pixels.

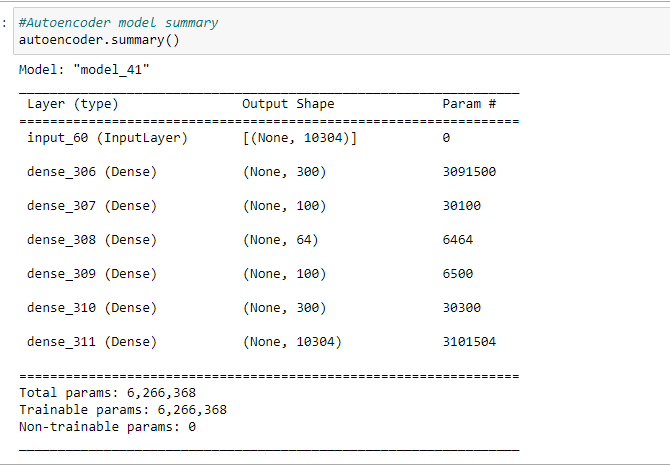


# 3. Data Preprocessing

In order to balance the dataset, the team created an **autoencoder** to produce new data samples (Images) and complete the groups with a low image count.

The initial step of the preprocessing is to reshape the data. Each image of size 120 x 92 pixels is transformed into a vector with 10304 features and its values are normalized between 0 and 255. After these preparations are complete the data can be fed into the autoencoder; its structure can be seen below.





After the creation of new images, each person (image group) has 48 images, increasing the total size of the dataset to 960 images for 20 people. An example of the results of the autoencoder can be seen below by comparing the input image, and the resulting (predicted) image.

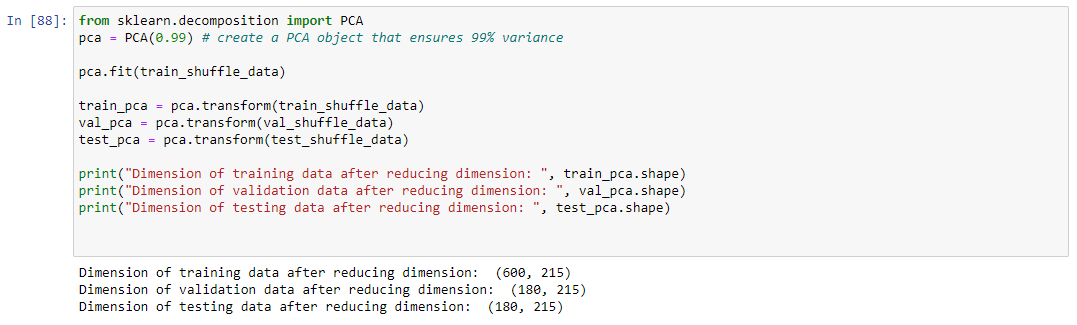


For the training, validation, and testing datasets, the original input was divided and shuffled in the following manner:

|  |  |  |
| --- | --- | --- |
| Total Images | 48 per person (Total (48 x 20) = 960) | 960 x 10304 |
| Training | 30 Images per Person (Total (30 x 20) = 600) | 600 x 10304 |
| Validation | 9 Images per person (Total (9 x 20) = 180) | 180 x 10304 |
| Testing | 9 Images per person (Total (9 x 20) = 180) | 180 x 10304 |

## 3.1 Dimensionality Reduction

For the final step of the preprocessing, and reducing the dimensionality of the data, the team selected to use PCA ensuring 99% variance of the data.



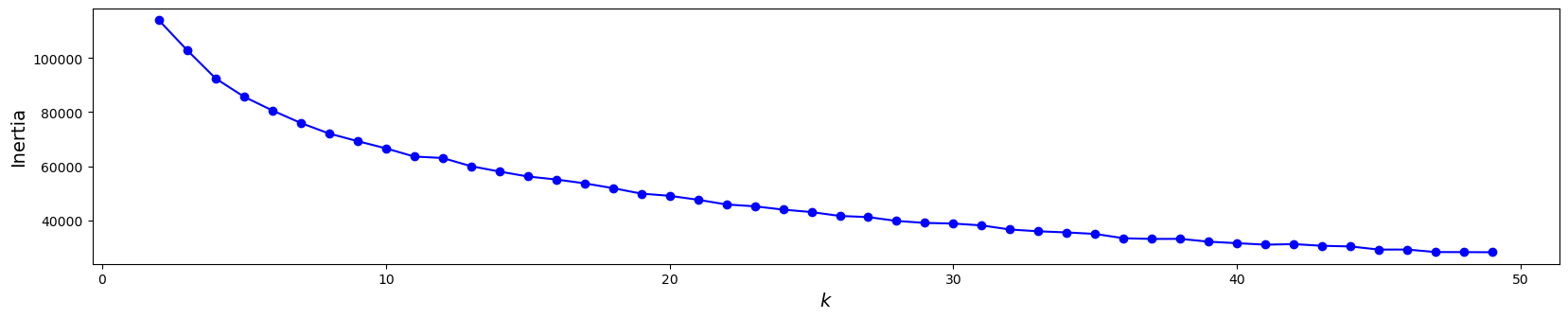
As the final result, it is noticeable how PCA is able to reduce the size of the vector representing each image from **10304 to 215** features.

# 4. Data Clustering

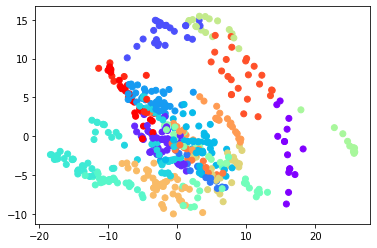
To frame the project under as an Unsupervised Learning problem, the team tried three different clustering algorithms in order to determine the one best suited to the facial images we received as input dataset. The objective of this section if to create the actual labels of the data by means of a clustering algorithm, and also confirm that the optimal number of clusters is consistent with the initial data exploration.

## 4.1 K-Means Clustering

For this algorithm, the team simply used an iterator to create simulations with a different number of clusters and finally determining the number of clusters by using the elbow plot below.



After the selection of 20 as an appropriate number of clusters, below is the plot of the approximate cluster in the dataset.

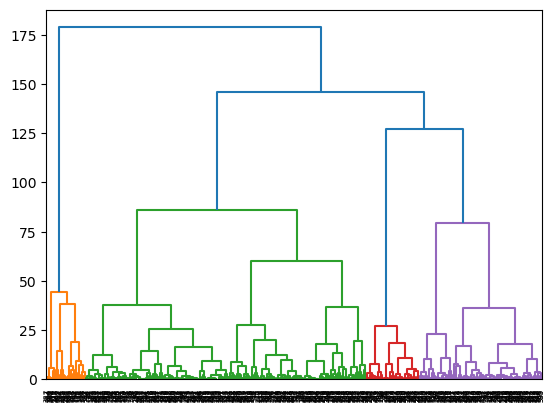




In order to push the algorithm to better determining the clusters, the k-means++ initialization method was used.

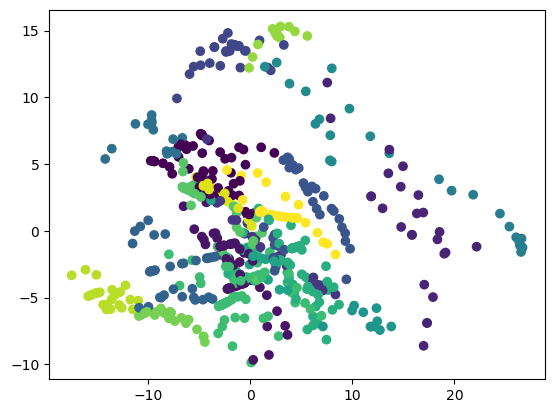
## 4.2Agglomerative Hierarchical Clustering

For the agglomerative hierarchical approach, the team started by using a dendrogram to determine the structure of clusters in the data.



After this initial exploration, the team used Agglomerative Clustering to determine the number of clusters by using ward linkage, the same used for the dendrogram, and Euclidean affinity.

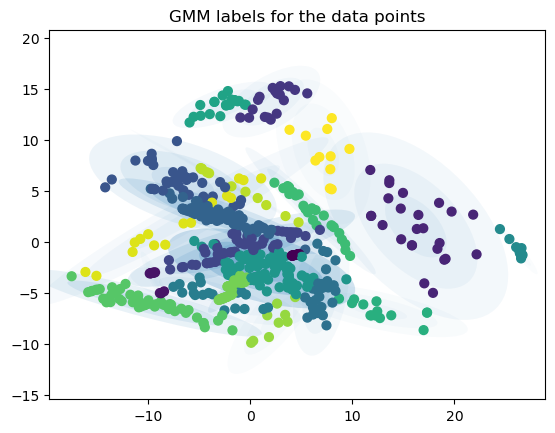




Similar results from k-means are obtained, however the shape of the clusters signals ellipsoidal clusters rather than spherical.

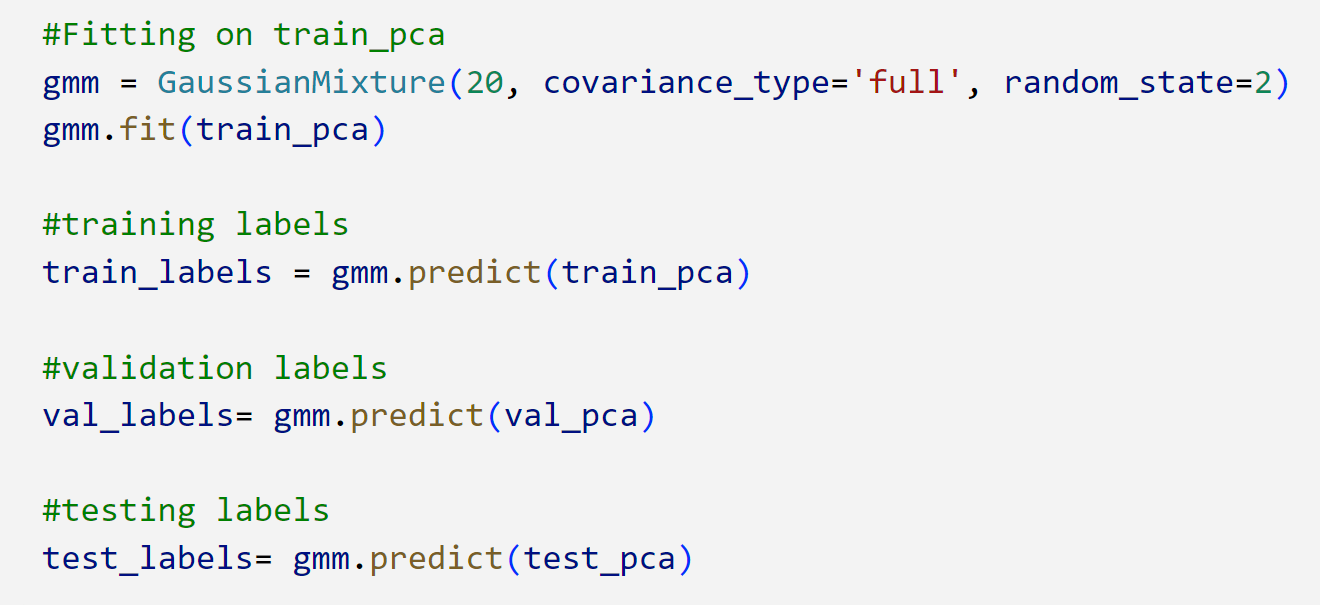
## 4.3 Gaussian Mixture Clustering

As the final of this stage of the project, Gaussian Mixture Clustering is used to determine the number of clusters considering it a better algorithm after the findings of previous steps and comparison with the multiclass classifier result of the next project stage.



In the previous figure it is possible to see how the shapes of the clusters are better defined by GMM with a more natural grouping.

After validating the results and confirming the compatibility with 20 clusters, GMM was applied over the data to produce the labels for all datasets: training, validation, and testing.



GMM works better when clusters shape is ellipsoidal because it used gaussian mixture and covariance rather than only depending on the center of the cluster. For the particular case of the project, GMM is generating less skewed data and also the same number of labels for each person.

By running the classifier described in the following section; using labels generated by all clustering techniques we found higher accuracy consistently when we used labels generated by GMM.

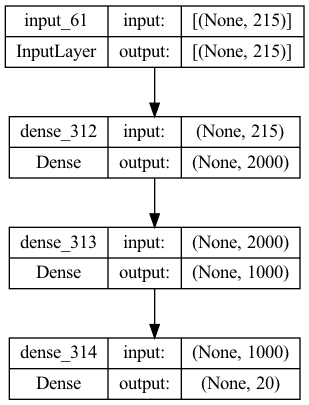
# 5. Multiclass Classifier – ANN

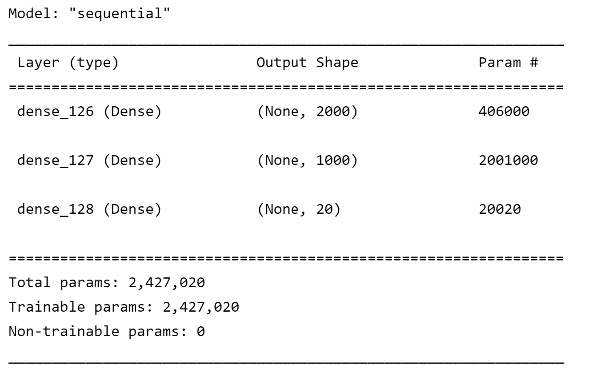
In the final stage of the project, after preprocessing the data, reducing the dimensionality of the problem and also using unsupervised learning techniques to create the data labels, all that remains is to create a Neural Network model to classify images and producing the expected results of identifying a particular face as one of the 20 individuals received as input.

For the multiclass classifier the team used leakyReLU as activation function to prevent the model to get stuck with 0 gradients and to give an additional effort in learning the fine features of the dataset. For multiclass problems, we also configure the model with ‘Softmax’ activation for the output layer and ‘Sparse Categorical Crossentropy’ for the loss with ‘Adam’ optimizer.



The previous image shows the details of the classifier; an ANN with two hidden layers. For the project, the input shape is determined by the results of PCA dimensionality reduction that produces 215 input features.

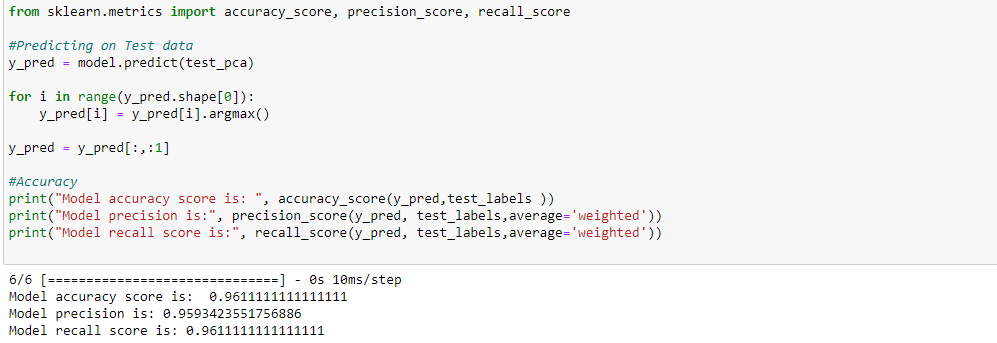




The output layer consists of 20 neurons, one for each class (person) to be identified by the model.

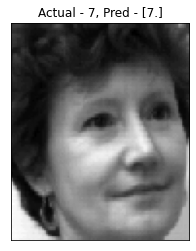
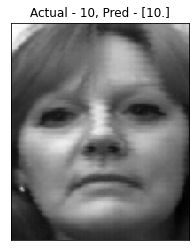
# 6. Model Results

For obtaining a consolidated result for the entire model, the teams created results for the precision and recall metrics using the weighted average method. Result for accuracy were produced as well.



As mentioned before, the combination of GMM and the simple yet robust architecture of the ANN classifier yielded very good results in this case. The precision and recall scores are good, showing how the model performs good on average for all cases and is able to correctly predict the required class.

Finally, in the following image the team has included a few samples of predicted images for closing the results section of the report.



# 

# 7. Conclusions

This project was created as a training ground and real-life application of the main concepts learned in the Supervised Learning section of the course. With direct experimentation, and even with with a small data set it is possible to obtain good results and appreciate the techniques inner workings first-hand.

Each technique and algorithm used in the project fulfilled its purpose and allowed to create a sophisticated model that is conformed of specialized components. Dimensionality reduction showed how much less data is required compared to the initial input, and it also shows that even in its reduced form, the model is able to understand and learn what is required from the data.

The use of PCA and GMM in the project proved incredibly useful by allowing the classifier model to be simpler compared to the usual architecture of more advanced models used for Image Processing with Deep Learning such as CNN and GAN.

The results obtained showed that the basic blueprint followed in the project can be used at scale with a larger dataset, and will allow to create a more complex model and effectively support law enforcement in the identification of people. With the results obtained additional use cases can be integrated into the model pipeline with an autoencoder capable of highlighting certain physical traits, removing accessories such as glasses and even the more useful case of analyzing thousands of images if a few minutes and identifying a person of interest without any human assistance.