ABSTRACT

Traffic sign recognition is done using machine learning models like convolutional neural networks(CNN’s) and fully connected neural networks(FCN’s) by which we recognize which traffic sign is there on a particular road.

Here we have taken GTSRB dataset and implemented CNN and FCN and trained the model to do the image recognition.

INTRODUCTION

Traffic sign recognition is a critical aspect of autonomous driving and has been the subject of intense research in recent years. Traffic signs are crucial for ensuring road safety and providing information to drivers, pedestrians, and cyclists. The German Traffic Sign Recognition Benchmark (GTSRB) dataset is a widely used benchmark for evaluating the performance of different algorithms for traffic sign recognition. In this report, we present an approach for traffic sign recognition using a Convolutional Neural Network (CNN) and a fully connected neural network(FCN) on the GTSRB dataset.

METHODOLOGY

A CNN is a type of deep learning model that is particularly well-suited for image recognition tasks. The architecture of a CNN consists of multiple layers of convolutional, max-pooling, dropout, dense, flatten ,batch normalisation.

In the second model we use a FCN with multiple layers of flatten, dense, batch normalisation and dropout.

In our approach, the input to the network was an image of a traffic sign resized to 30x30 pixels. The output of the network was a vector of probabilities for each of the 43 classes of traffic signs in the GTSRB dataset. The model was trained using the Adam optimization algorithm with a cross-entropy loss function.

CNN: -

A Convolutional Neural Network (CNN) is a type of deep learning neural network that is designed to process data with grid-like topology, such as an image. CNNs have been proven to perform exceptionally well in various computer vision tasks, such as image classification, object detection, and segmentation.

The main building blocks of a CNN are:

Convolutional Layers: these layers perform a mathematical operation called convolution on the input data, which extracts relevant features from the data.

Pooling Layers: these layers reduce the spatial resolution of the data, making the network more computationally efficient and also reducing overfitting.

Activation Functions: these functions introduce non-linearity into the network, allowing it to learn complex representations.

Fully Connected Layers: these layers are responsible for making the final prediction, based on the learned features from the previous layers.

One of the key strengths of CNN is that they can learn hierarchical representations of data , meaning that they can learn simple features in the early layers and combine them to form more complex features in the deeper layers. This allows CNN to learn increasingly complex and abstract representations of the data , leading to improve performance on a wide range of computer vision tasks

In conclusion, Convolutional Neural Networks are a powerful tool for processing grid-like data and have In conclusion, Convolutional Neural Networks are a powerful tool for processing grid-like data and have been proven to perform exceptionally well on various computer vision tasks. They have become a popular choice in the field of deep learning, and their widespread use has led to many breakthroughs in the field.

FCN: -

FCN (Fully Convolutional Network) is a type of neural network architecture commonly used in computer vision tasks such as semantic segmentation and object detection. Unlike traditional CNNs, FCNs are designed to take in an image of any size and produce an output of a corresponding size by processing the image in a dense, per-pixel manner.

FCNs achieve this by introducing "skip connections" that concatenate high-level features from earlier in the network with low-level features from later in the network. This allows for a rich fusion of feature information from different parts of the network, leading to improved accuracy in the output.

FCNs have achieved state-of-the-art performance on several benchmark datasets for tasks such as semantic segmentation and object detection, and have been widely adopted in industry and research.

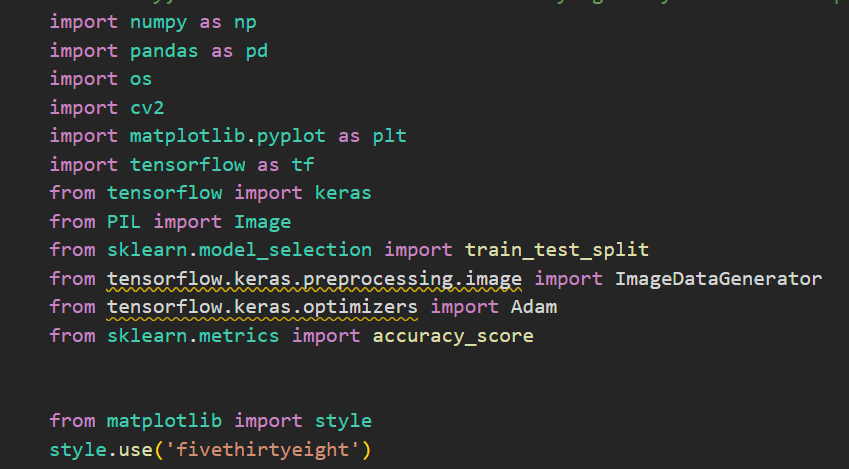
One of the key advantages of FCNs over traditional CNNs is their ability to process inputs of arbitrary size, making them well suited for tasks that involve image data of varying sizes. Additionally, the dense, per-pixel prediction of FCNs makes them well suited for tasks that require fine-grained analysis of an image, such as semantic segmentation.

Overall, FCNs are a powerful tool for computer vision tasks and have proven their effectiveness in a range of applications.

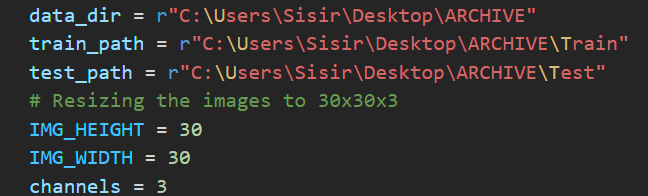
CODE EXPLANATION: -

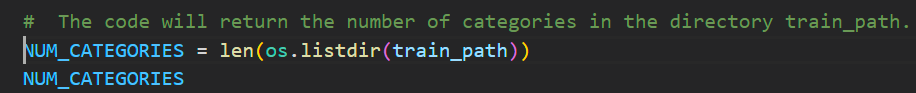
Part 1: -(CNN)

Import all the libraries



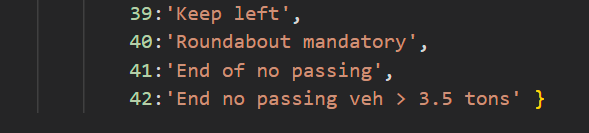
Load the data set and image resize parameters



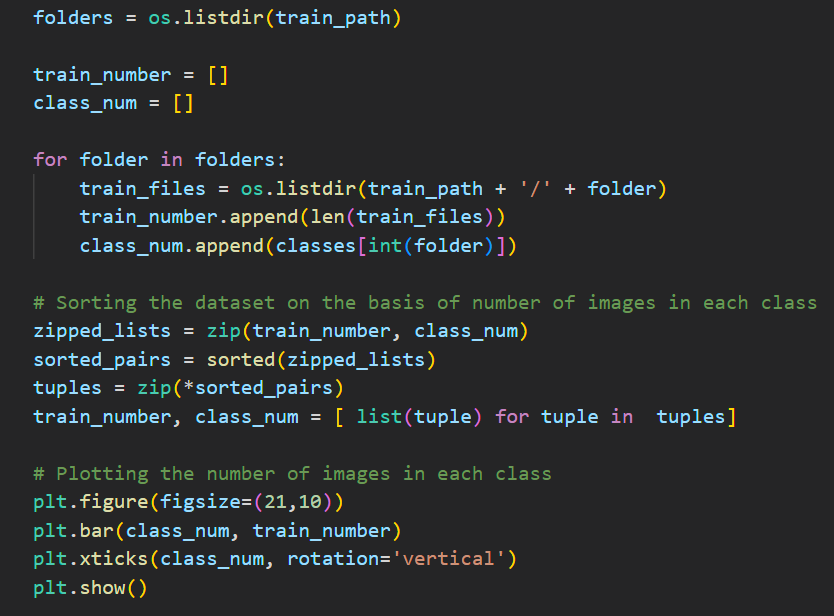
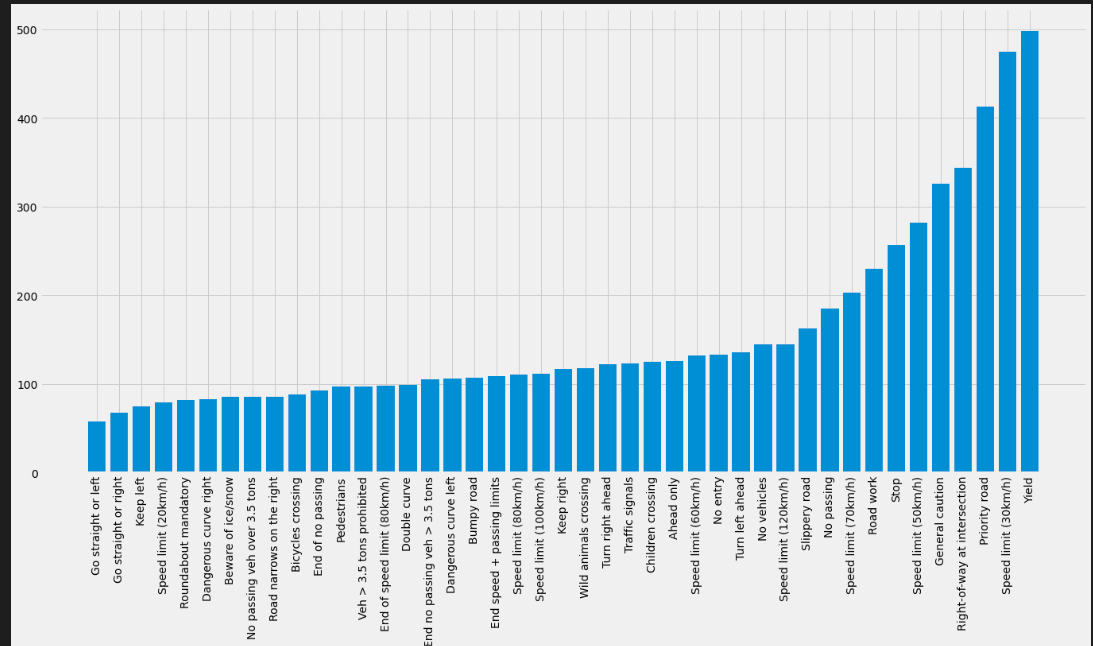


Declare the class labels: -

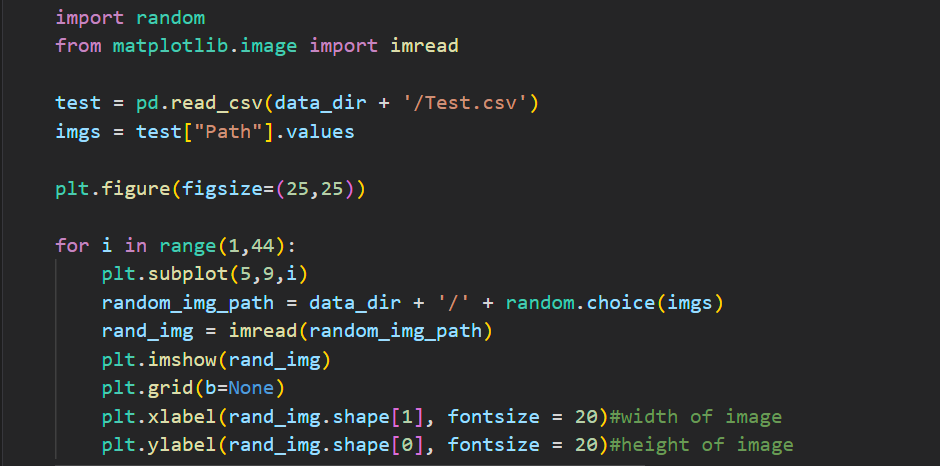




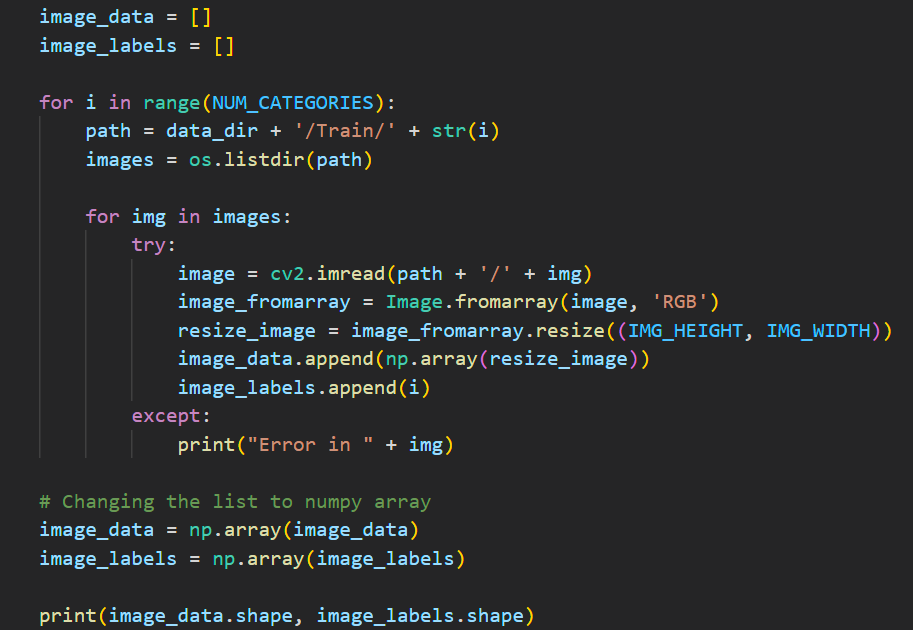
Plot a bar graph showing no of images in each class in the ascending order: -

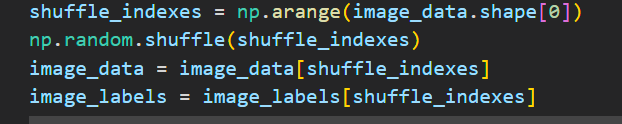
 

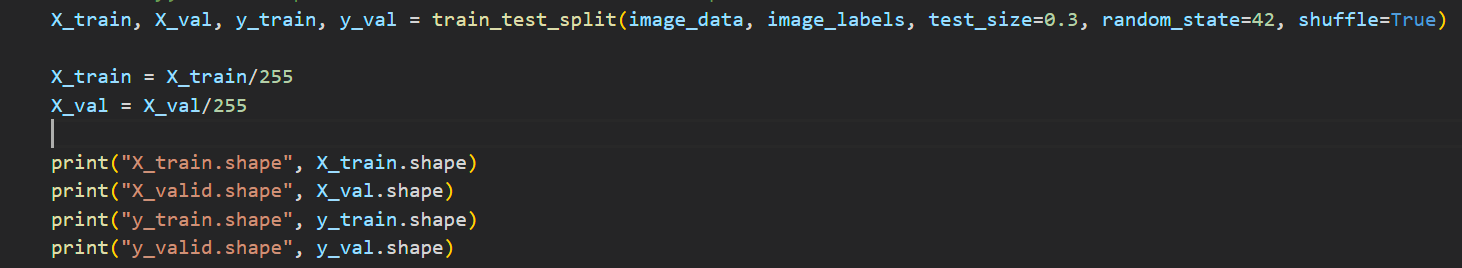
Visualize the images in the dataset

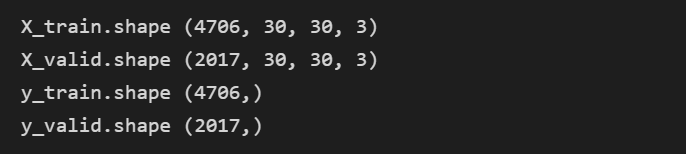
 

Resizing the images from dataset with the parameters given above

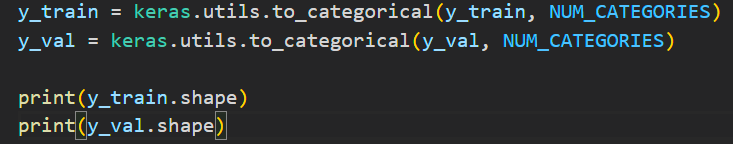
 

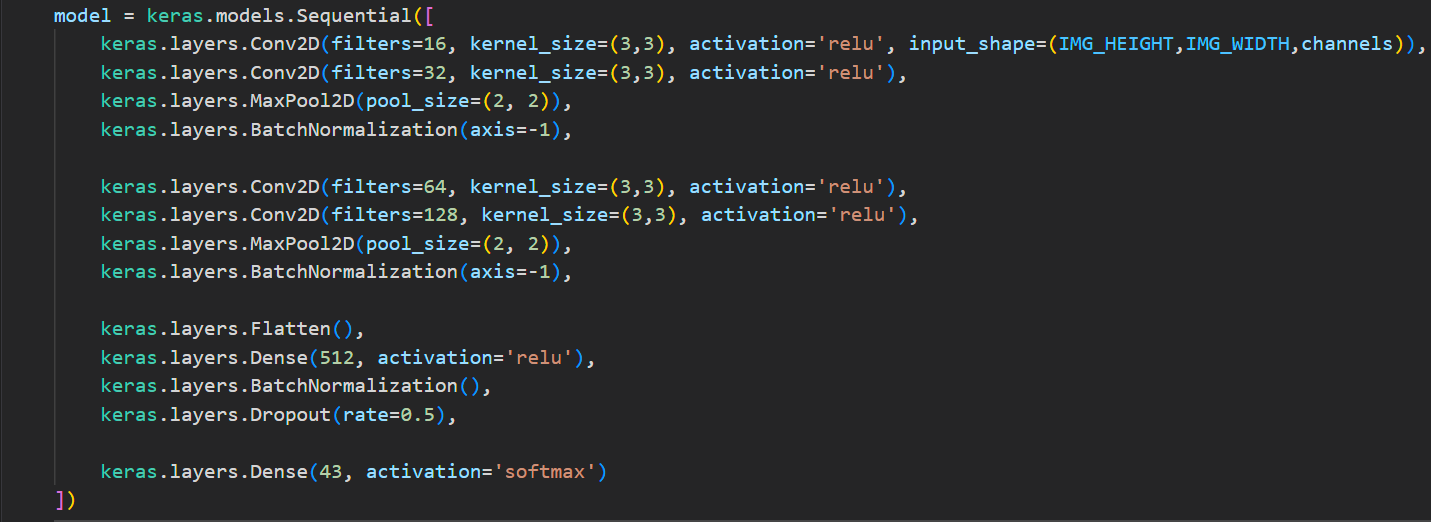
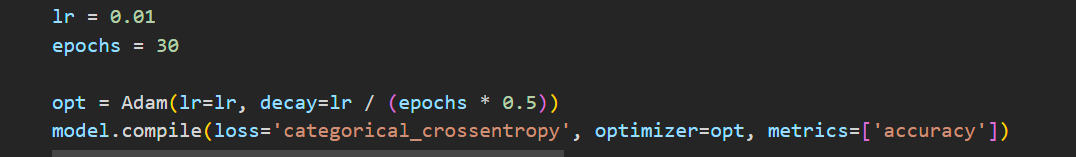
Randomly shuffle the indices of the image data 

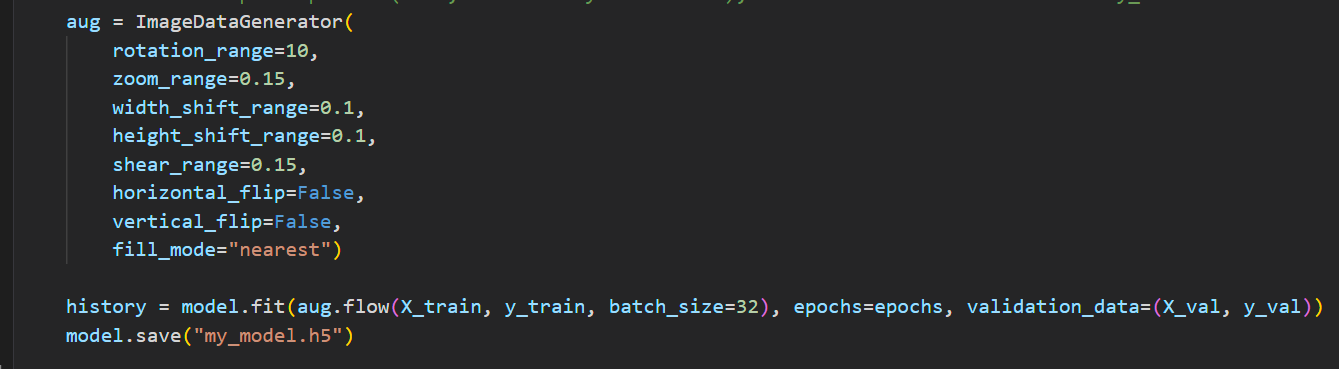
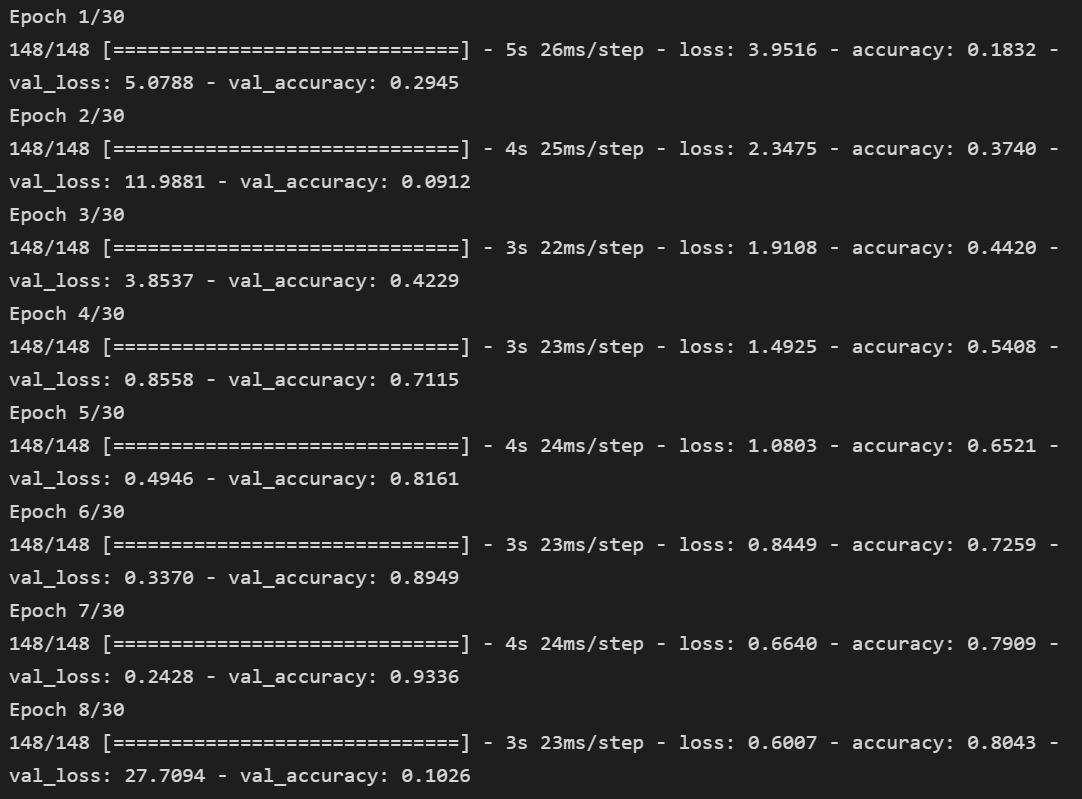
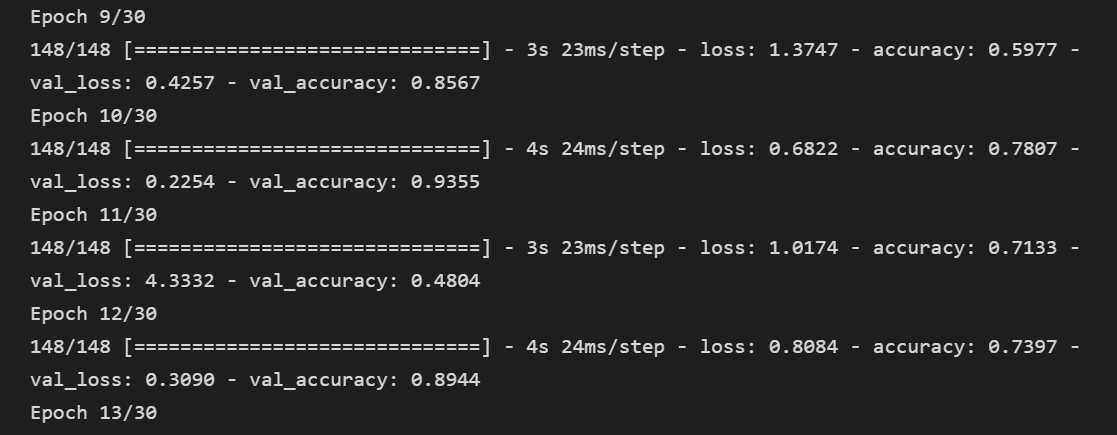
Splitting train(70%) and test data(30%) and printing their shape 

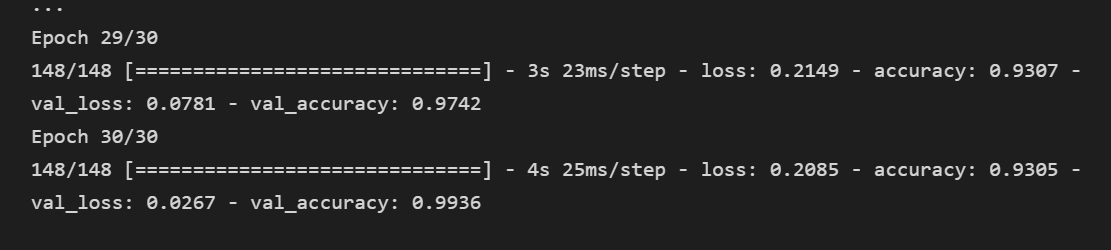


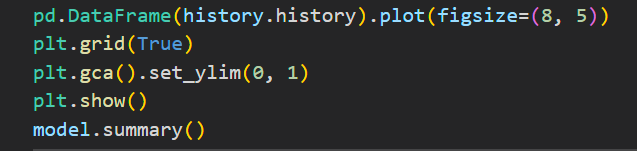
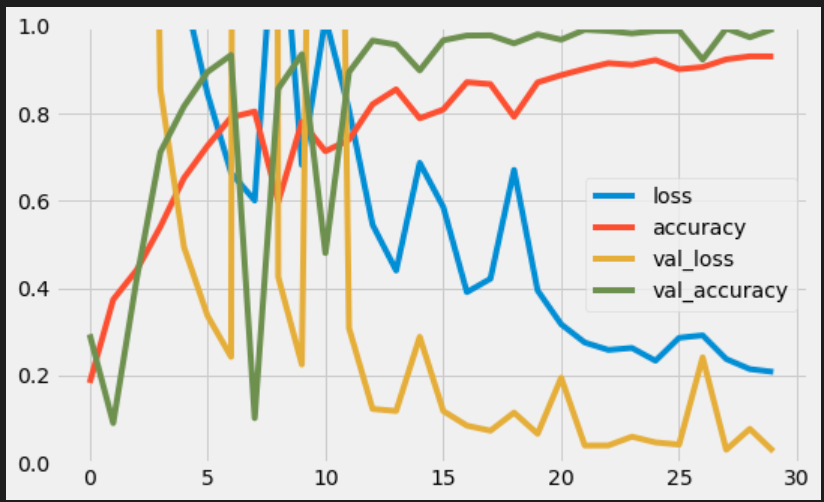
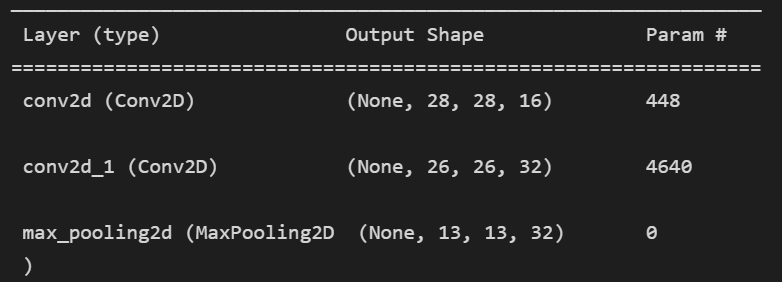
convert a 1D array of integers (**y\_train** and **y\_val**) into a categorical representation

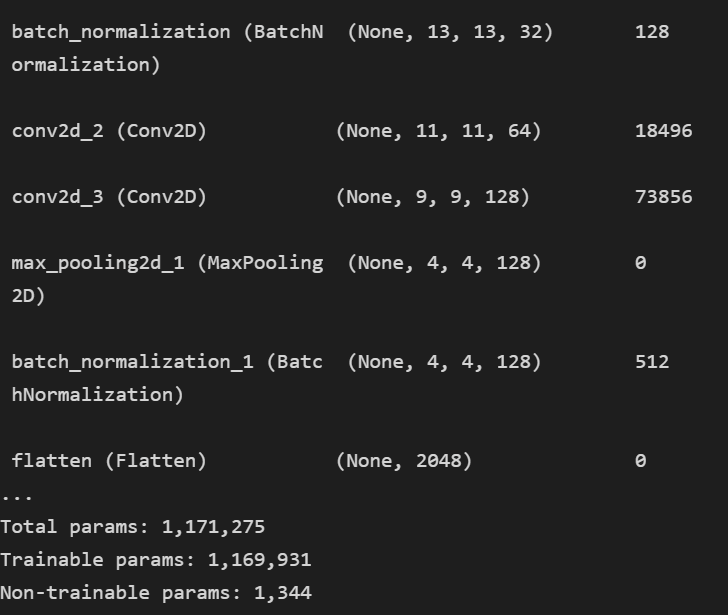


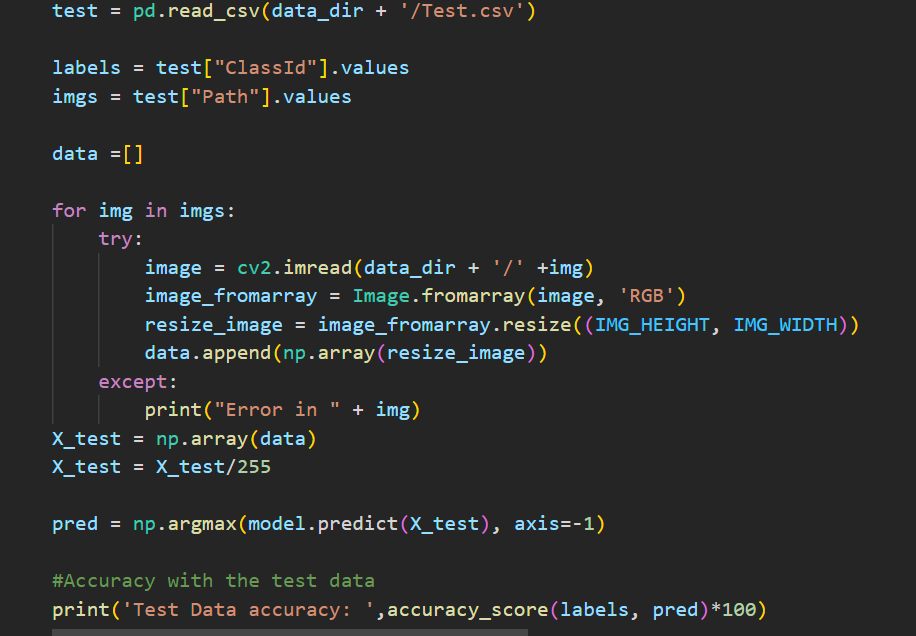
Create the CNN model by adding layers   
specifying the learning rate, no of epochs and compiling the model 

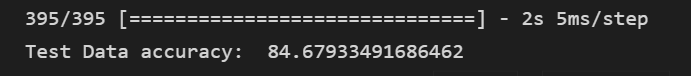
Performing image augumentation and fitting the data into model and running it.   



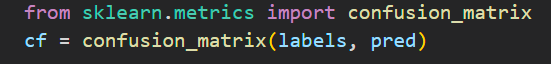
Plotting the loss and accuracy graphs and model summary   

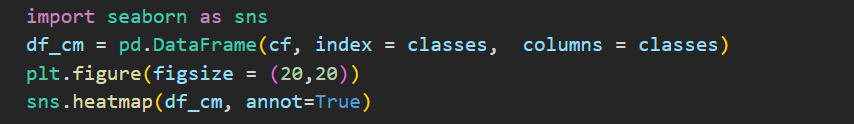
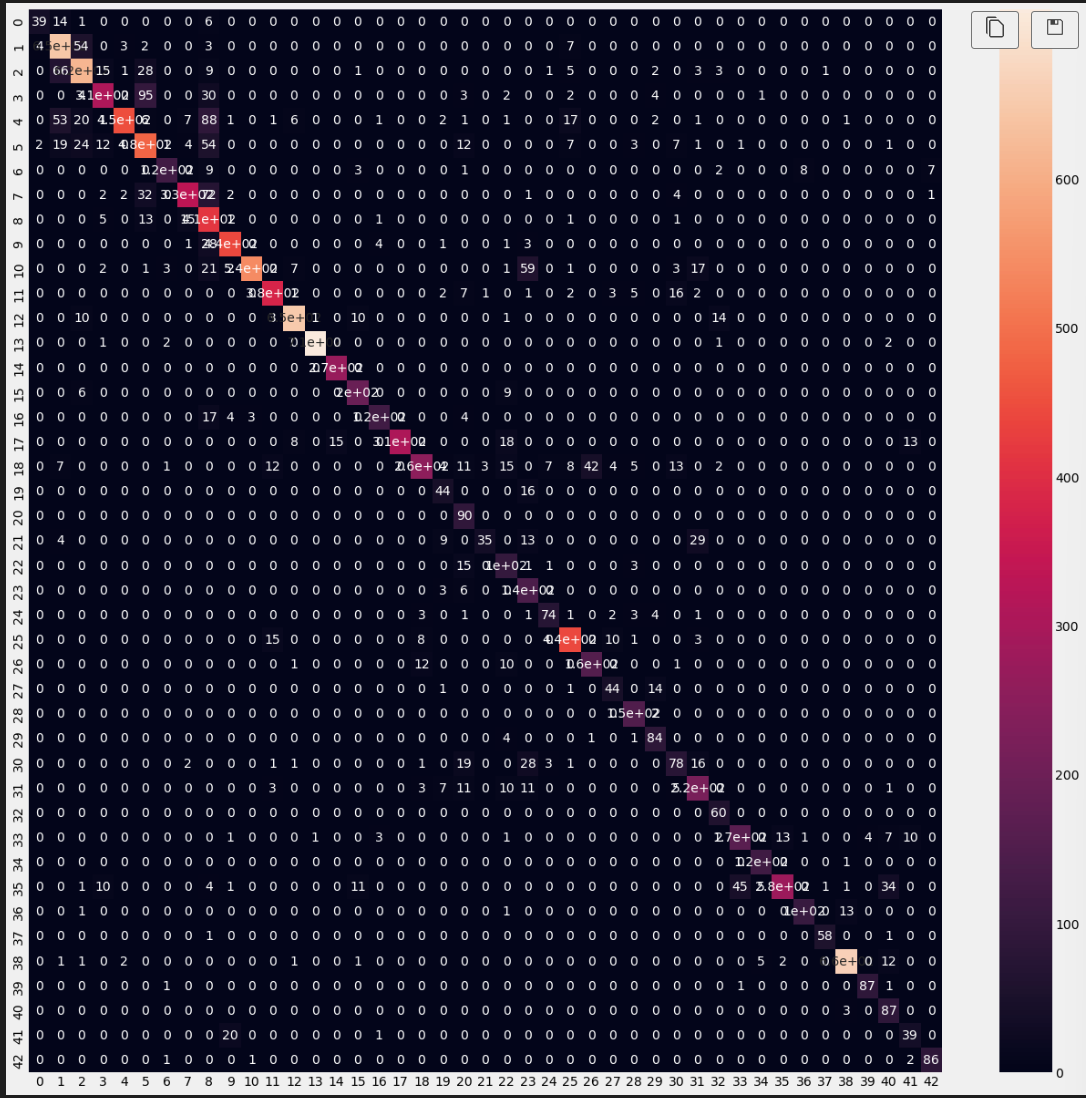


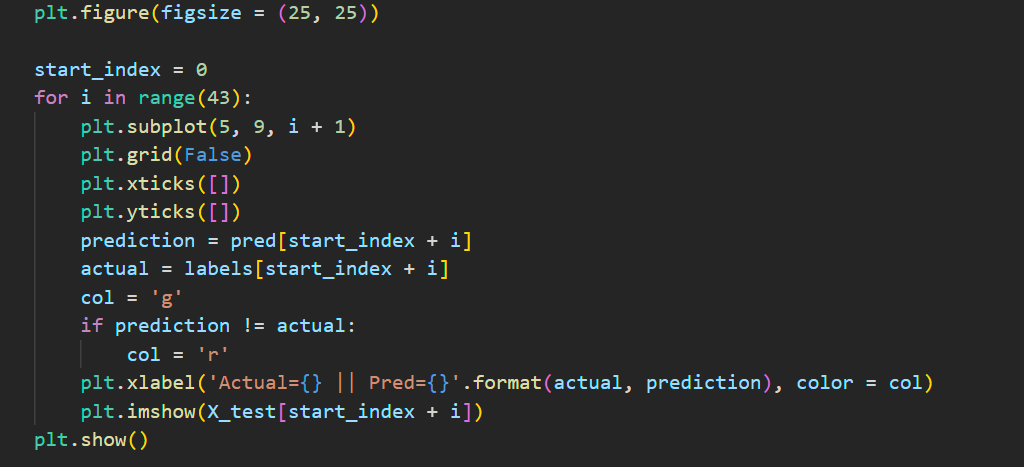
Printing the test data accuracy 

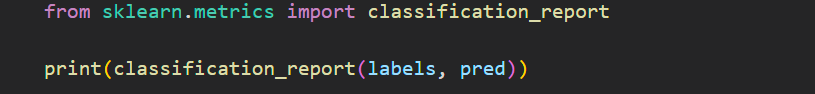
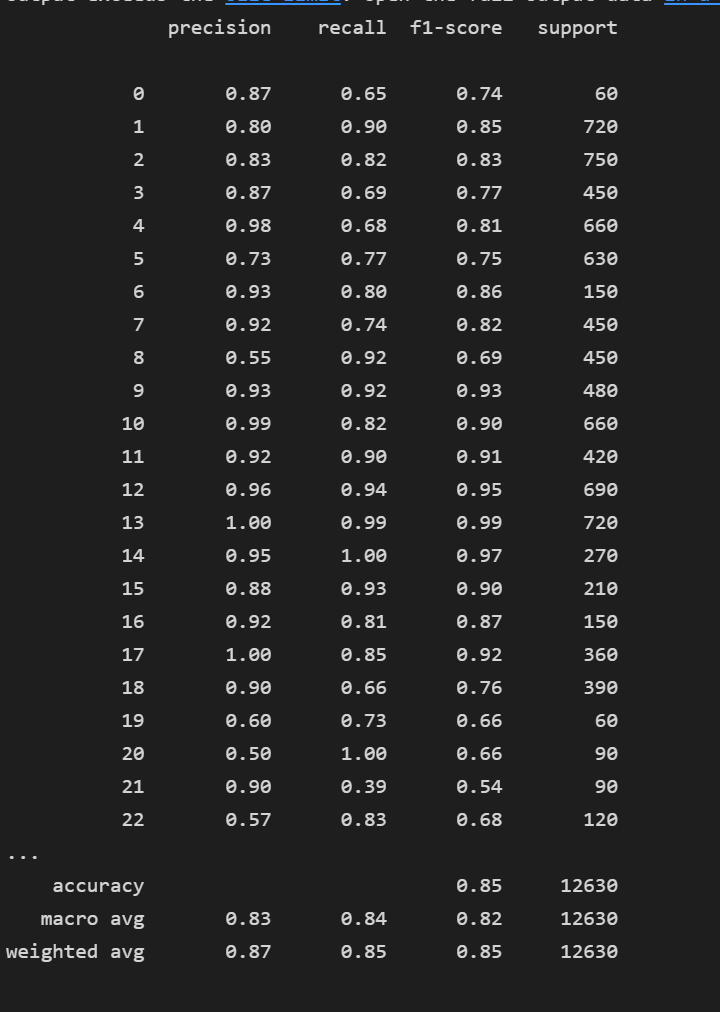


Create a confusion matrix



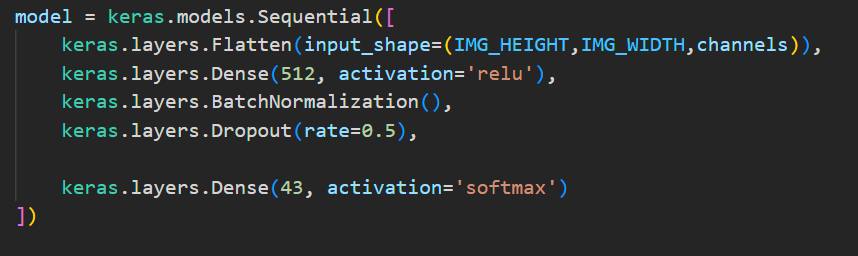
Print the heatmap  

Plot images and comparing the prediction and real values of the class predicted by the model  

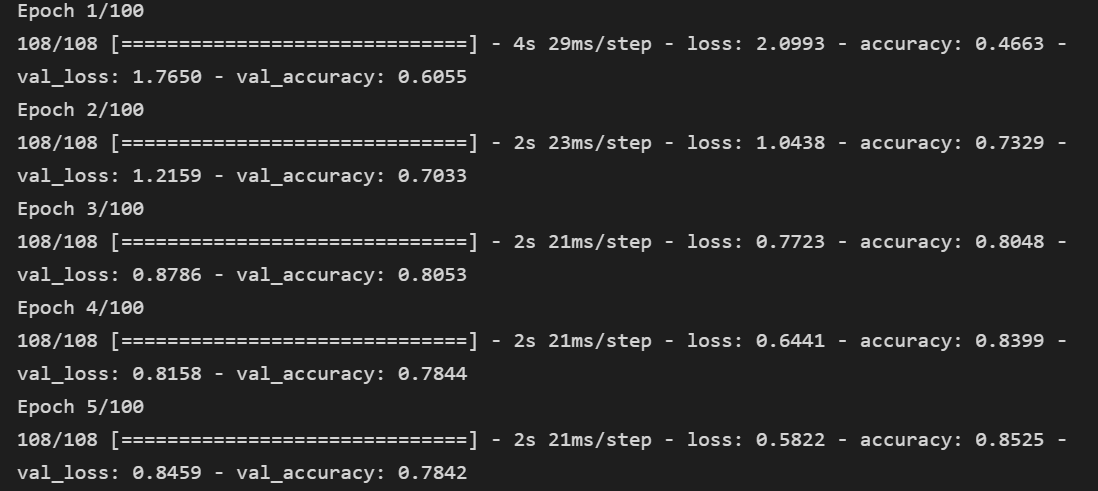
Print the classification report  

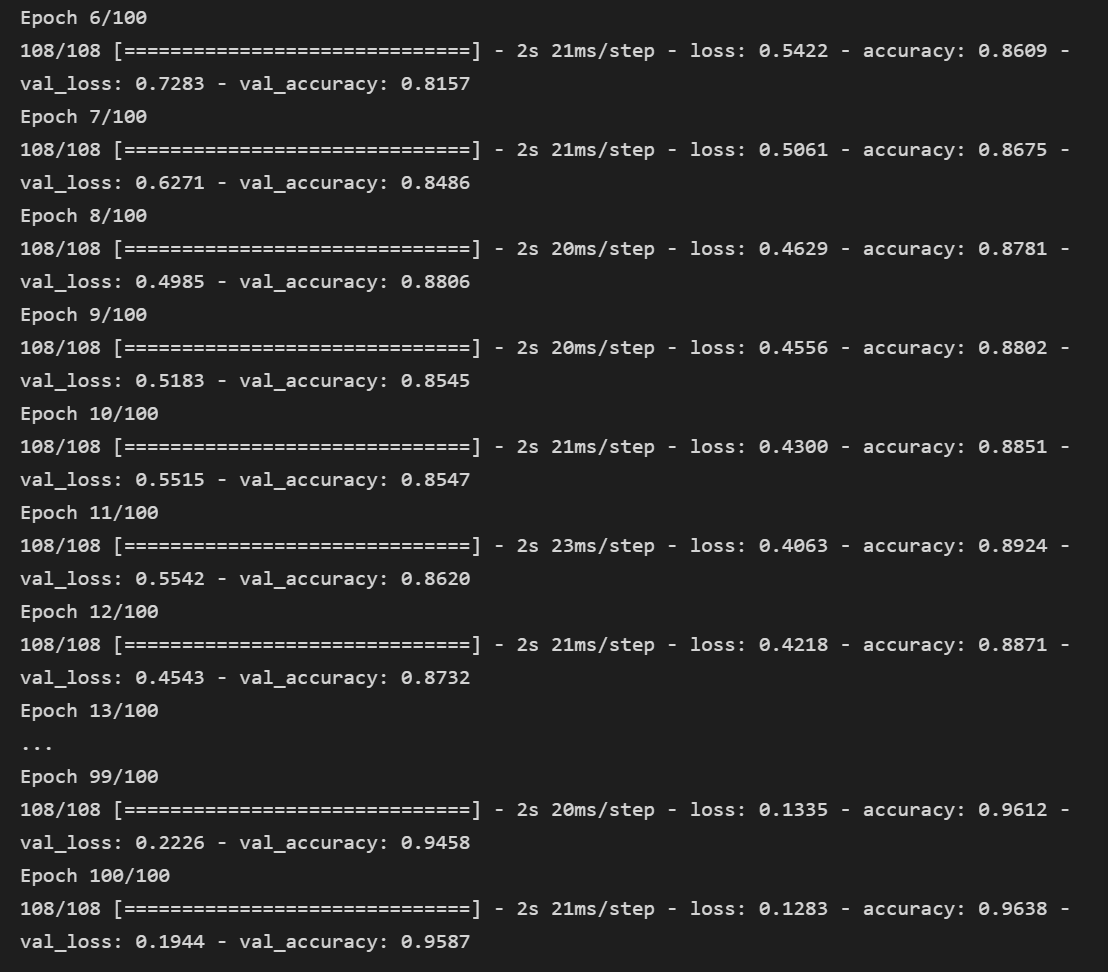
Part 2: - (FCN)

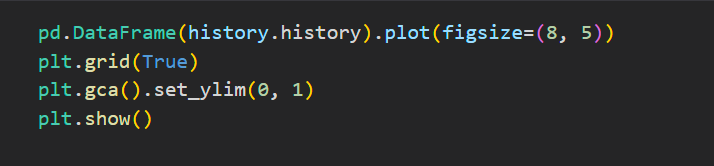
In FCN we take a larger dataset containing 39209 images and a different neural network architecture apart from these every operation that we do to the dataset is same . so we will start with the model and see its performance

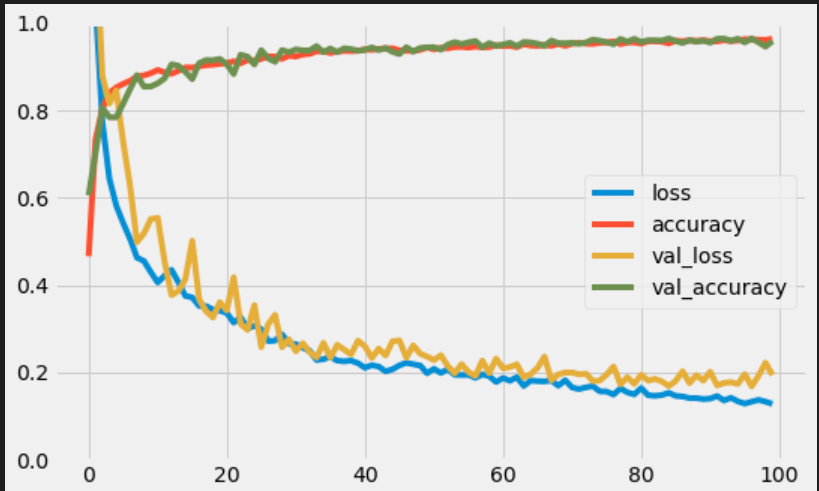


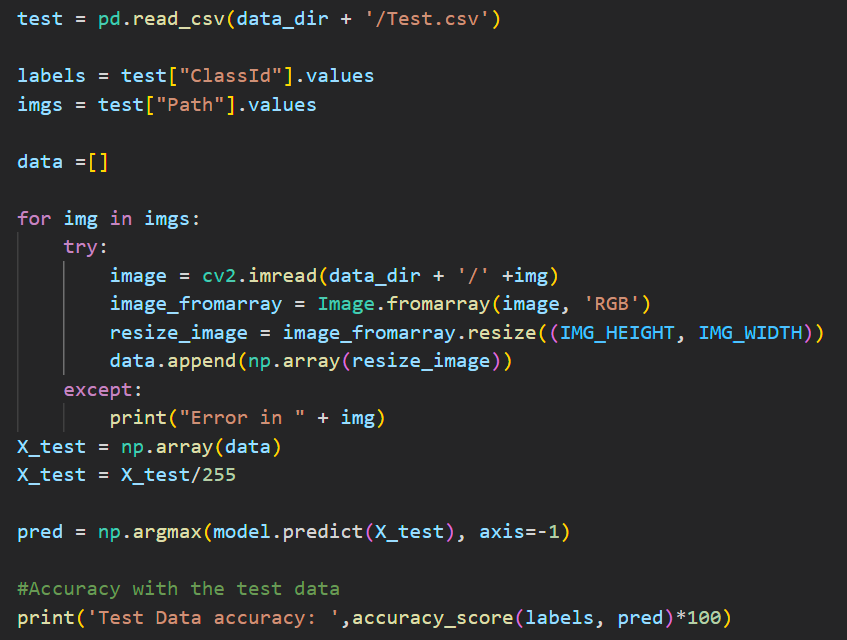
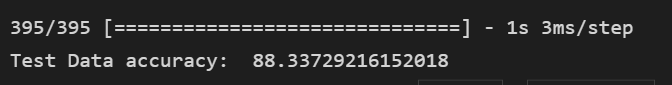
Specifying how the learning rate decreases with each epoch and compiling model and fitting the dataset into the model and running it 



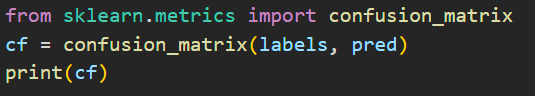


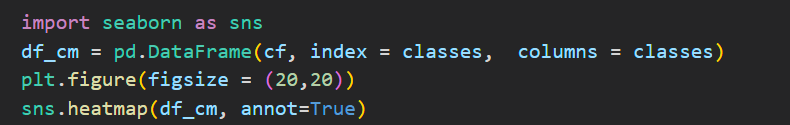
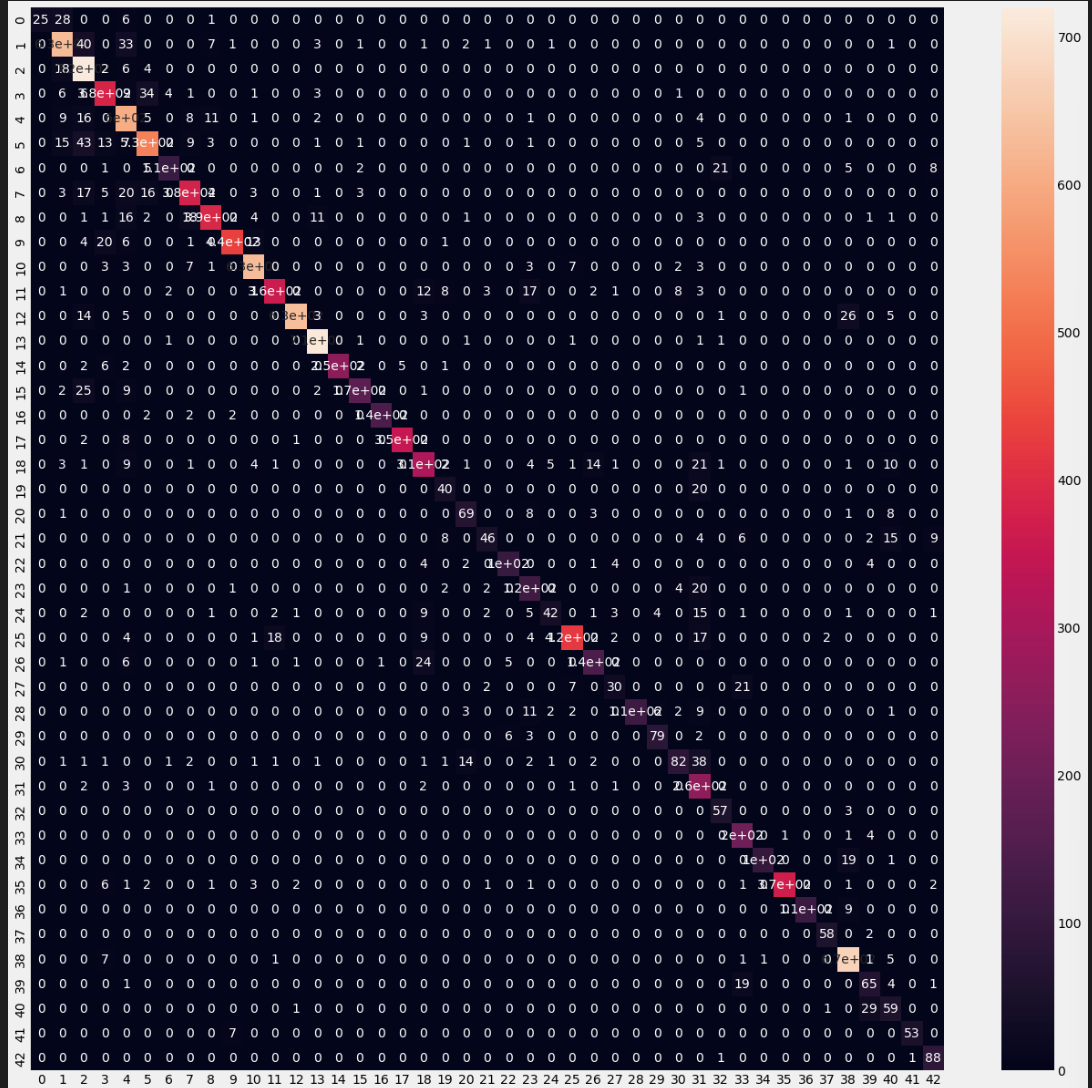
Plotting the accuracy and loss graph 

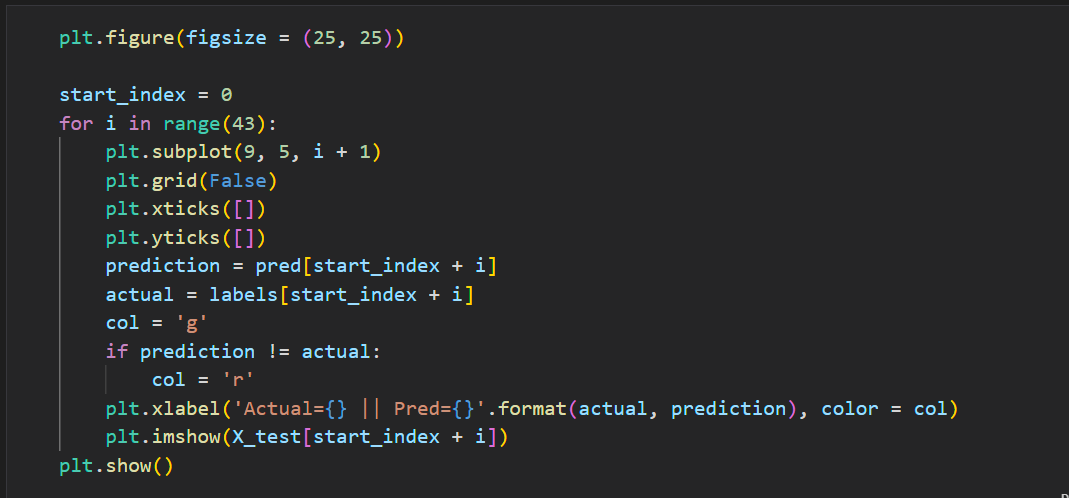


Printing the test data accuracy  

Creating the confusion matrix



Plotting the heatmap  

Plotting the images and comparing the predicted and real values of the labels predicted by the model.



CONCLUSION

The results of our study demonstrate the feasibility of using CNNs and FCN’s for traffic sign recognition on the GTSRB dataset. The high accuracy achieved by the model shows that CNNs can effectively recognize traffic signs in the GTSRB dataset. However, further studies are necessary to assess the generalization of the model to different types of road signs and environments.

FUTURE WORK

There are several directions for future work that can build upon our approach. For example, further experiments can be performed to evaluate the robustness of the model to changes in lighting conditions, occlusions, and other factors that can affect the performance of traffic sign recognition in real-world scenarios. Additionally, it would be interesting to investigate the use of other deep learning models, such as Generative Adversarial Networks (GANs) or Recurrent Neural Networks (RNNs), for traffic sign recognition. Another area of interest is improving the generalization of the model to new types of road signs that are not present in the training set. This can be done by using data augmentation techniques, such as rotation, scaling, and flipping, to increase the size of the training set.