Pattern Recognition and Machine Learning Minor Project Topic: Using MLP,RandomForest and SVM to classify whether a person is wearing a mask

M K Laksath (B19CSE045), Himanchal Sharma (B19CSE039)

I. Introduction

Machine learning is becoming a more integral part of society every day as more and more systems grow to depend upon data and feedback to further optimize or change. One of the most popular forms of data under more than exponential growth is the image/video data. The data containing pixels or pixels per frame in case of videos.The amount of information that can be gained from these pixels even for just a small 32x32 image is tremendous, and how much a set of just 10,000 images can help you recognise,organise or predict is also beyond bounds. In times such as these various models with mind blowing mathematics and tricky practical fixes are becoming the trend, models keep getting better, smarter and faster and so do the methods of making and using them. There are three models designed to categorise a certain kind of object from an image namely: Support Vector Machine, Multi Layer Perceptron,RandomForestClassifier. We attempt to do a comparative study of these three models based on various parameters to judge the best model for predicting on the dataset.

II. Multi-layer Perceptron

Multi layer perceptrons were the predecessors of the present day popular Convolutional Neural Networks. These were the first runs at trying to mimic the structure of the neurons of the brain and impart the model the ability to learn the best representation of the input data and find the most suitable way to relate it to output, in a way to generate a mathematical mapping between the two. MLP is a single layer neural network that is precursor to multi layer more complex models.

*1) **Methodology**:*

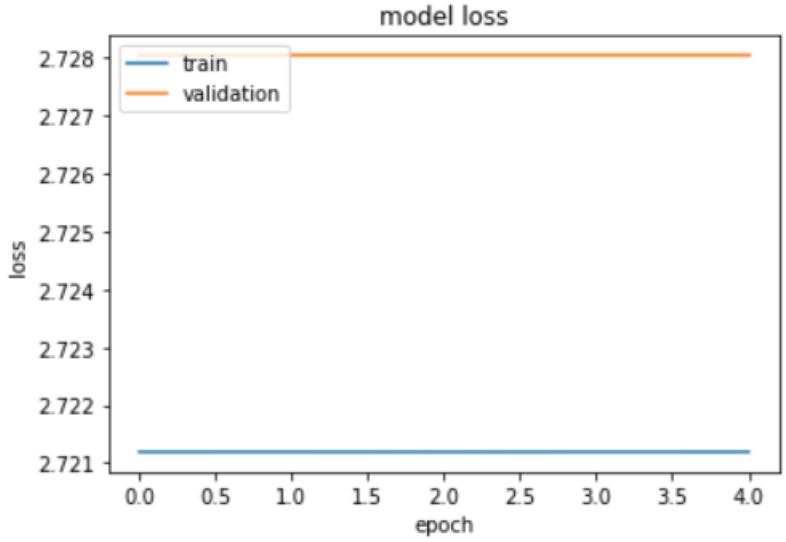
1) Principle: What the MLP tries to achieve is make a network of neurons with multiple layers where each layer contains multiple neurons with their own weights,biases and activation function(each layer has one). The idea is to assign a random set of weights biases and a chosen activation function to neurons in layers and then adjust them throughout the training process to best map them to the output.
2) Specifications: Before running this dataset through the model all the images from different shapes were converted into one (128,128,3) format so the models can process them. Then again the images were reshaped into (imagenumber,49152) format to feed into the MLP Neural Network.Also we discovered that as much greyscaled and hog form images did better on other models and even on the sklearnMLP, it performed critically bad on my custom made perceptron (greyscale hogged images gave almost 0 accuracy after 5 epochs).
3) Building: For the custom made perceptron: A sequential model was chosen as those are the preferred type of models with plain layers containing exactly one input and output tensor. Two 'dense' type of layers were added to the network. Dense layers are the most common form of layers which perform the following function on the input: output = activation(dot(input, kernel) + bias) And returns the output. Here for the first two layers, the number of neurons were chosen to be 256 with the activation function chosen as the most common type "Rectified Linear Unit" function which "is a piecewise linear function that will output the input directly if it is positive, otherwise, it will output zero." The first layer also contains the dimensions of the input the layer will receive which will be Batchsize,49152 where batch size will be given at the time of running.The final layer contains ten neurons and the activation function is chosen as softmax.

$$\sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$

Softmax Activation:

Even though sigmoid is the function generally recommended for binary classifications, in this case it only achieves an accuracy score of 0.09 so we decided to go with softmax as the output layer activation function. Next we've defined the SGD optimizer as the optimizer for our network with the standard learning rate of 0.01 and momentum of 0.9. We have chosen nesterov to be true What is Nesterov: "Much like Adam is essentially RMSprop with momentum, Nadam is Adam RMSprop with Nesterov momentum." The model was compiled to use SGD as the optimizer with binarycrossentropy as

the loss function to be minimised and valaccuracy to be maximized. The model was trained for 5epochs i.e ran over the entire dataset 5 times and the chosen loss over epochs was plotted as follows:



A. Conclusion

The final loss and accuracy score over the test set were found as for the custom MLP : [2.709165334701538, 0.9047834277153015] And [0.8305452572565332] for the SK learn classifier

III. Random Forest Classifier

Random forest is based on the idea that "A large number of relatively uncorrelated models (trees) operating as a committee will outperform any of the individual constituent models." That is what it aims at. Perhaps one of the oldest and first of its kind ensemble classifiers, random forest is very popular at classifications of text and image data and performs quite well on popular datasets like iris and MNIST.

*1) **Methodology**:*

1) Principal: What random forest aims to do is take a number of weak decision tree classifiers and make an ensemble decision by assigning separate weights to outputs of those weak classifiers and compensate for weaknesses of individual trees. During the training process these assigned weights are tuned to get the best possible accuracy.
2) Specifications: Before running this dataset through the model all the images from different shapes were converted into one 128,128,3 format so the models can process them. Then the images were converted into grayscale and then were hogged in order to reduce the computing power required by reducing dimensions of the features while retaining maximum data variation. We discovered that hogging improves non neural network model's performances by 2 to 8 points at least, generally.
3) We used Sklearn random forest classifier for our data where criterion entropy performed better than gini by a score of 0.01

A. Conclusion

Increasing or varying any other parameter even by a huge number only seemed to affect the model by a score of 0.01 and the peak we receive by applying gridsearch was at 0.84 for randomforest on the test set.

IV. Support Vector Machines: SVM

SVM aims to draw a decision boundary through linearly separable classes. When there are many lines possible to draw and differentiate between classes, We take the line that brings out the maximum margin between them.

Support vectors are just points that lie nearest to the decision boundaries. Support vectors are the ones that maximize the decision boundary margin.

Hard margins are the ones that allow no points to fall inside the decision margin. Whereas, soft margins allow some slack.That is, they allow some points to fall inside the margin.This is useful as it allows some class overlaps which could be needed in free cases. "hard margin" because there are no points that fall inside the decision margin. In many cases of SVM classification, it makes more sense to use a "soft margin", which would give "slack" to some points and allow them to encroach on the dividing region

$$h(\beta) = \frac{1}{2}\|\beta\|^2 + C\left[\frac{1}{n}\sum_{i=1}^{n}\max(0, 1 - y_i(\beta x_i))\right]$$

Here , a smaller value of C will lead to a softer margin .This formula is called hinge loss function.
Upon minimizing this function, we get a set of betas that look similar to linear regression function.

$$\hat{y}_i = \begin{cases} -1, & \beta_0 + \beta_1 x_{i1} + \cdots + \beta_k x_{ik} < 0 \\ 1, & \beta_0 + \beta_1 x_{i1} + \cdots + \beta_k x_{ik} \geq 0 \end{cases}$$

Prediction are made in the following way:

*1) **Methodology**:*

1) Specifications:Initially, the images are not in order. Therefore, the dimensions of the images were reconstructed and made 32x32x3. To improve the accuracy again, the coloured image was converted to grayscale images. Upon this process, the grayscale image is not suitable for conversion into hog image. Hog images are a type of preprocessing step where the image is converted to an outline-based sketch which results in the highlighting of the most significant pixels in the array. Then this image is used as our input data to our SVM classifier.
2) Building: GridSearch: The decisionfunctionshape was set to be [’ovo’, ’ovr’]. The value of ’C’ was set to be [0.1,1,10]. The ‘kernel’ values were chosen to be [’linear’, ’rbf’, ’sigmoid’]

A. Conclusion

1) The best parameters:’C’: 10, ’decisionfunctionshape’: ’ovo’, ’kernel’: ’rbf’ The maximum accuracy obtained : 0.9356180486481599.

V. Contributions

1) Himanchal Sharma: Studying and implementation of MLP and RandomForest based approach.
2) M K Laksath : Support Vector Machine with implementation, analysis and report and documentation as well as combining codes