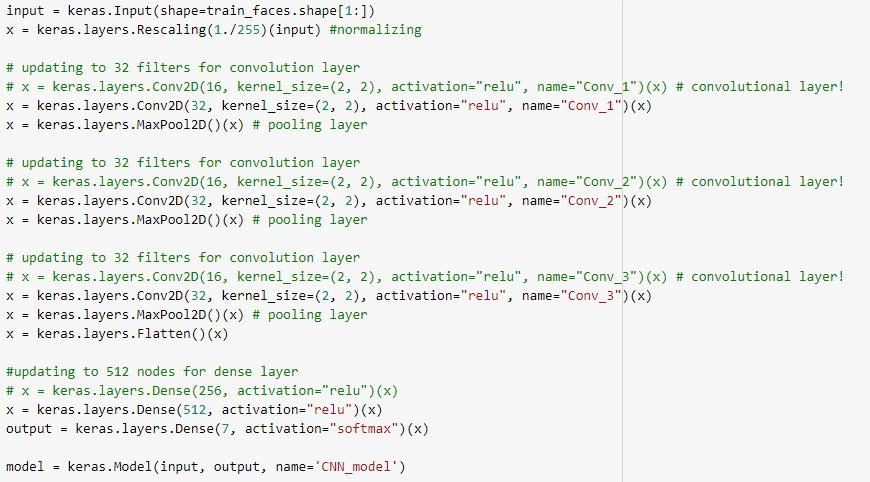
Tim Miller

15.S04.SP22

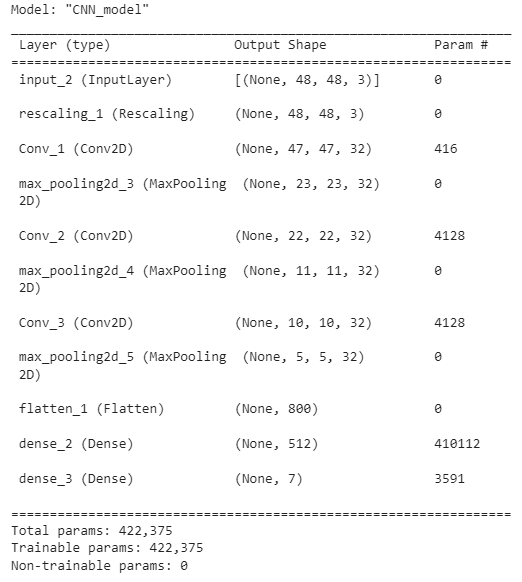
Homework #1

**PROBLEM 1A**

Updating to 32 filters for convolution layers and 512 nodes:

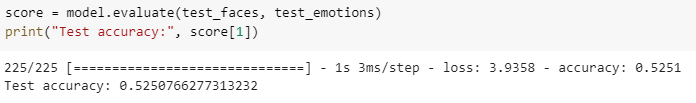


After these updates, we get **422,375 parameters**.

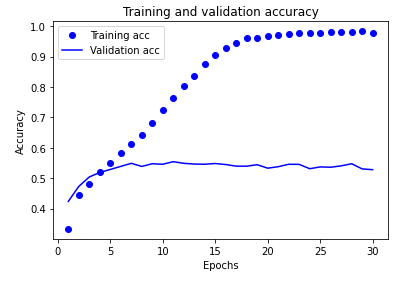


**PROBLEM 1B**

We find a model accuracy of ~52.51%



**PROBLEM 1C**



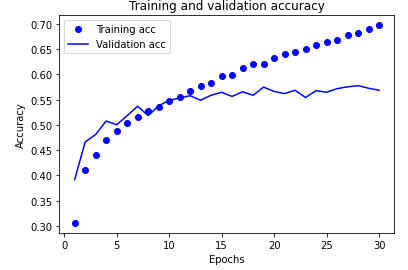
We can see the training accuracy approaches 100% as the number of epochs increases. However, validation accuracy reaches an asymptote of ~55%. Likely the model is overfitting on noise in the training set, which is why the accuracy is not replicated in the validation set.

**PROBLEM 2A**

Deep learning models require a lot of data to be trained. Data augmentation is useful because it helps us create a larger data set to train our models without necessarily having to go out and find additional images of faces (or whatever we are training our models). In this case we flipped the image and zoomed in / out providing additional data for training. But we did it with the data we already had.

**PROBLEM 2B**

Below is a graph for accuracy vs. epochs for the CNN augmented model.



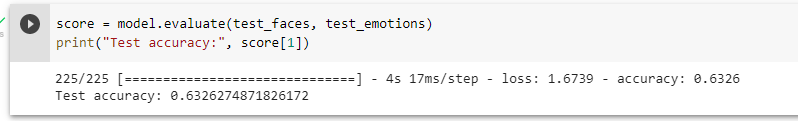
**PROBLEM 2C**

Compared to problem 1C, we see two things

* The training accuracy in 2B does not increase in accuracy as rapidly as 1C
* The difference between the training and validation accuracy lines is small

The likely explanation for this that data augmentation helps reduce overfitting. As we said in 1C, the reason the training accuracy approaches 100% is because the model is likely overfitting. That is why it does not perform as well on the validation set. But in 2B, overfitting is reduced, so the model does not increase in accuracy as rapidly. Then accordingly the accuracy for the training and validation models is closer.

**PROBLEM 3A**



The accuracy for this model is ~63.3%.

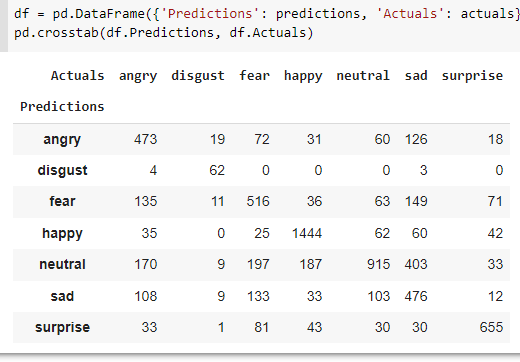
**PROBLEM 3B**

Transfer learning is useful here because we have access to VGG19 which has been trained on the >1million images in ImageNet. Given the size of this dataset, and the fact it is much bigger than the set of images we are analyzing, we are comfortable that VGG19 is not at risk of overfitting. Furthermore, VGG19 has been trained for image recognition and feature classification – like the end use case we are running with our sample data.

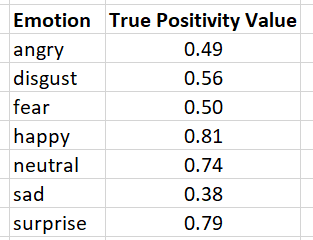
Transfer learning can be necessary when the you have a small amount of data in the sample you are testing. Given deep learning models require a large amount of training data, your results will likely not be great with small sample of data you have. Instead, you can leverage the work already done with another model and tweak the weights using your sample data. In many cases, this allows you to achieve impressively high accuracy levels even with a small amount of data. However, as stated above we need to make sure the model used for transfer learning has much more data than what we are testing to prevent overfitting.

**PROBLEM 3C**

Confusion matrix



If we looked at the true positivity, we find the following



Clearly, the values for negative emotions like anger, disgust, fear, and sadness are much lower than positive or neutral emotions. The simplest answer would be to add in additional datapoints (images) for angry, disgust, fear, and, sad emotions to train the model better. Or we could find a transfer learning model that is specifically good at looking at negative emotions to add to the transfer learning model we already built.

**BONUS**

I successfully loaded my image into the notebook but encountered an error when I ran the prediction. I think this has something to do with the way my image is formatted as I enter it into the prediction model.

