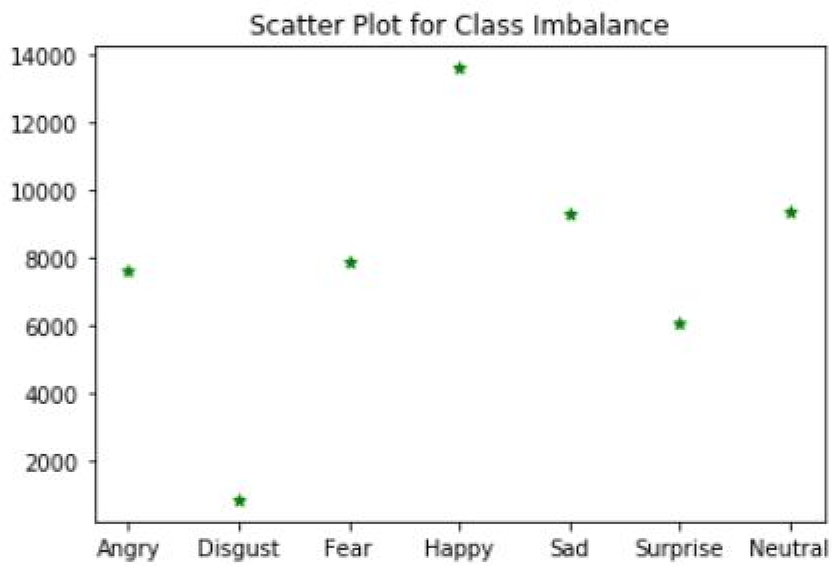
Affective Computing | Assignment 1 (Coding)

Divyanshu Kumar Singh | 2017048

1.) A brief text about data, its visualization, its classes, whether data is balanced:

The provided dataset contains 54597 grayscale images of 48*48 dimensions of different facial emotions. The emotions are classified into 7 classes as follows: 'Angry', 'Disgust', 'Fear', 'Happy', 'Sad', 'Surprise', 'Neutral'. Number of images in each class has following distribution: **[7586 -> Angry, 835-> Disgust, 7868-> Fear, 13587-> Happy, 9284-> Sad, 6071-> Surprise, 9366-> Neutral]**. This is represented using the scatter plot below:



Using this scatter, we can clearly see there is a class imbalance present in the dataset.

2.) Why have you chosen this particular model or technique for classification?

I have chosen a basic deep learning approach using a very small neural network with three layers. The sole reason behind opting for this approach was, images dataset works really well with CNN compared to other Machine Learning model taking into account the dataset size. Here we had enough data set for applying into a small CNN model. Also, a benefit of using CNN over other classification technique is that you need not worry about the feature extraction. The model summary is shown below:

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_6 (Conv2D)            (None, 46, 46, 64)        640
_________________________________________________________________
max_pooling2d_6 (MaxPooling2 (None, 23, 23, 64)        0
_________________________________________________________________
conv2d_7 (Conv2D)            (None, 19, 19, 128)       204928
_________________________________________________________________
max_pooling2d_7 (MaxPooling2 (None, 9, 9, 128)         0
_________________________________________________________________
flatten_3 (Flatten)          (None, 10368)             0
_________________________________________________________________
dense_3 (Dense)              (None, 7)                 72583
=================================================================
Total params: 278,151
Trainable params: 278,151
Non-trainable params: 0
_________________________________________________________________
```

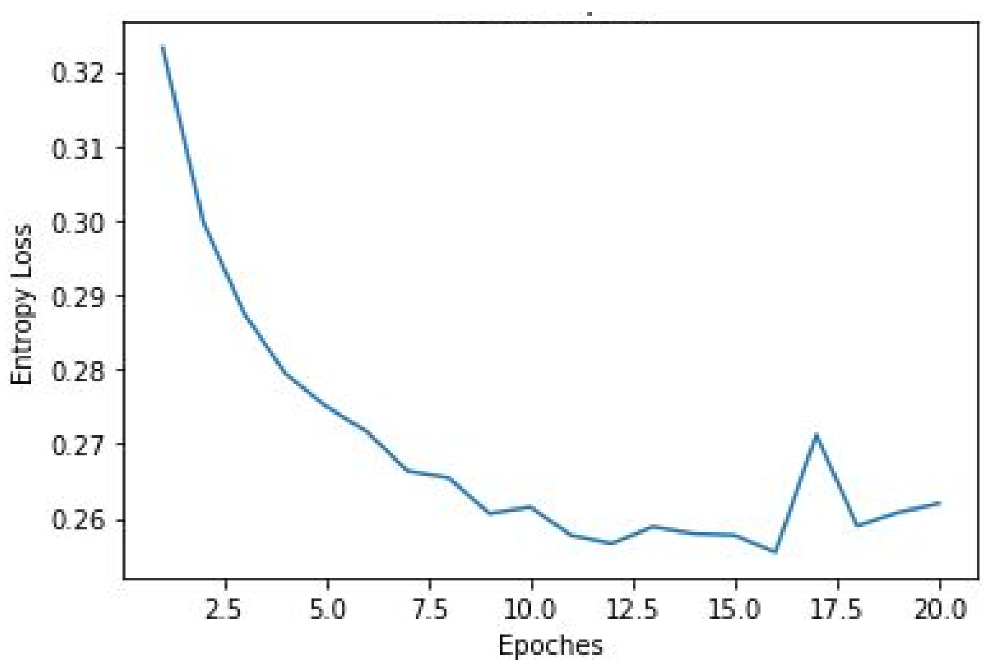
3.) Analysis of Result :

I split the data into an 80:20 ratio, i.e 80% for training and rest 20% for testing. This is quite a common approach using learning techniques. We found an overall **Test Accuracy: 90.39%**.

I used 20 epoch for the training purpose, and outcome for every training epoch is shown below:

```
Train on 43677 samples, validate on 10920 samples
Epoch 1/20
43677/43677 [==============================] - 13s 300us/sample - loss: 0.3470 - acc: 0.8666 - val_loss: 0.3232 - val_acc: 0.8724
Epoch 2/20
43677/43677 [==============================] - 13s 293us/sample - loss: 0.3022 - acc: 0.8797 - val_loss: 0.2997 - val_acc: 0.8802
Epoch 3/20
43677/43677 [==============================] - 13s 293us/sample - loss: 0.2816 - acc: 0.8865 - val_loss: 0.2875 - val_acc: 0.8851
Epoch 4/20
43677/43677 [==============================] - 13s 289us/sample - loss: 0.2669 - acc: 0.8921 - val_loss: 0.2794 - val_acc: 0.8884
Epoch 5/20
43677/43677 [==============================] - 13s 291us/sample - loss: 0.2551 - acc: 0.8969 - val_loss: 0.2750 - val_acc: 0.8911
Epoch 6/20
43677/43677 [==============================] - 13s 292us/sample - loss: 0.2442 - acc: 0.9020 - val_loss: 0.2716 - val_acc: 0.8922
Epoch 7/20
43677/43677 [==============================] - 13s 289us/sample - loss: 0.2336 - acc: 0.9069 - val_loss: 0.2663 - val_acc: 0.8935
Epoch 8/20
43677/43677 [==============================] - 13s 291us/sample - loss: 0.2249 - acc: 0.9110 - val_loss: 0.2654 - val_acc: 0.8950
Epoch 9/20
43677/43677 [==============================] - 13s 288us/sample - loss: 0.2169 - acc: 0.9143 - val_loss: 0.2606 - val_acc: 0.8964
Epoch 10/20
43677/43677 [==============================] - 12s 285us/sample - loss: 0.2087 - acc: 0.9176 - val_loss: 0.2615 - val_acc: 0.8964
Epoch 11/20
43677/43677 [==============================] - 12s 284us/sample - loss: 0.2016 - acc: 0.9211 - val_loss: 0.2577 - val_acc: 0.8983
Epoch 12/20
43677/43677 [==============================] - 12s 282us/sample - loss: 0.1953 - acc: 0.9240 - val_loss: 0.2566 - val_acc: 0.8998
Epoch 13/20
43677/43677 [==============================] - 12s 282us/sample - loss: 0.1893 - acc: 0.9263 - val_loss: 0.2589 - val_acc: 0.8997
Epoch 14/20
43677/43677 [==============================] - 12s 280us/sample - loss: 0.1833 - acc: 0.9296 - val_loss: 0.2579 - val_acc: 0.9012
Epoch 15/20
43677/43677 [==============================] - 12s 280us/sample - loss: 0.1780 - acc: 0.9318 - val_loss: 0.2577 - val_acc: 0.9020
Epoch 16/20
43677/43677 [==============================] - 12s 280us/sample - loss: 0.1740 - acc: 0.9338 - val_loss: 0.2554 - val_acc: 0.9032
Epoch 17/20
43677/43677 [==============================] - 12s 280us/sample - loss: 0.1683 - acc: 0.9363 - val_loss: 0.2712 - val_acc: 0.9010
Epoch 18/20
43677/43677 [==============================] - 12s 281us/sample - loss: 0.1644 - acc: 0.9374 - val_loss: 0.2590 - val_acc: 0.9030
Epoch 19/20
43677/43677 [==============================] - 12s 280us/sample - loss: 0.1593 - acc: 0.9404 - val_loss: 0.2608 - val_acc: 0.9047
Epoch 20/20
43677/43677 [==============================] - 12s 281us/sample - loss: 0.1550 - acc: 0.9419 - val_loss: 0.2620 - val_acc: 0.9040
Test Accuracy: 90.39642810821533
```

The loss-vs-epoch graph is shown below:



4. Literature survey what is in trend in FER. It can be anything related to a new Dataset released or a new type of new technology that is popular. It can also include any new idea of what you think.

Facial Emotion Recognition (FER) is one of the most interesting research problems when it comes to affective computing and deep learning. With the advancement of deep learning methods, the FER has grown ever since. In 2013, at ICML Workshop in Challenges in Representation Learning a dataset/challenge was posted named Dataset: Facial Emotion Recognition and this was further taken onto Kaggle. A lot of researchers have worked on this dataset [1], and that best accuracy reacher so far is 71.162% using deep networks. Apart from this data, there exists a lot of such datasets deadline around with emotion recognitions examples MMI Facial Expression Database[3], Facial Expression in the Wild and etc. One of the states of the artwork proposed a CNN approach and were able to achieve significant accuracy on Extended Cohn-Kanade (CKP) and MMI dataset where state of art previous were 99.2% and 93.2% respectively and their model outperformed them by achieving 99.6% on CKP and 98.63% on MMI [2]. In my view, an interesting application could be to use in Human-Computer Interaction or Human-Robot Interaction, for examples, we can robot-assisted intervention for the diagnosis of various mental health disorders. An individual interacts with a social robot and fills a questionnaire and you can report about it using FER, this could actually help countries where the needs of special educator and doctors are quite less compared to the population suffering from such disorders.

References

[1] Goodfellow, I. J., Erhan, D., Carrier, P. L., Courville, A., Mirza, M., Hamner, B., ... & Zhou, Y. (2013, November). Challenges in representation learning: A report on three machine learning contests. In *International Conference on Neural Information Processing* (pp. 117-124). Springer, Berlin, Heidelberg.

[2] Burkert, P., Trier, F., Afzal, M. Z., Dengel, A., & Liwicki, M. (2015). Dexpression: Deep convolutional neural network for expression recognition. *arXiv preprint arXiv:1509.05371*.

[3] https://mmifacedb.eu/

[4] https://cs.anu.edu.au/few/AFEW.html

[5] https://arxiv.org/pdf/1704.06756.pdf