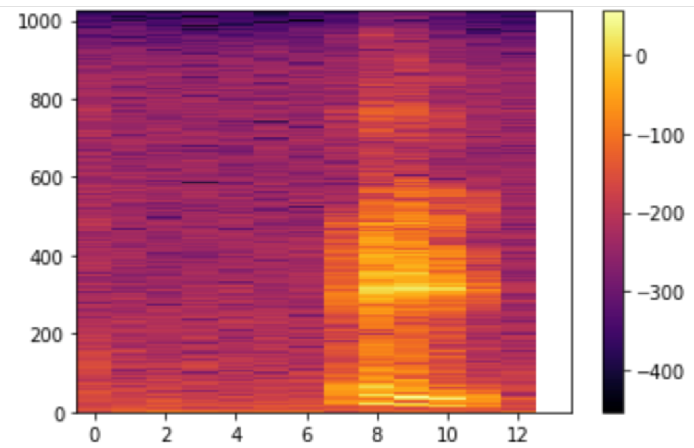
KSHITIJ GULATI
2017162

ASSIGNMNENT 2

QUESTION 1:

In question 1, we write an algorithm for generating spectrogram features from scratch and plotting them. The spectrogram image for a window size of 2048, overlap of 50% with 16000 sampling rate is as follows.




```
]: array([[ -28.42426984,  -29.80849052,  -30.41452629, ...,  -29.43195828,
          -32.43147751,           -inf],
        [ -51.20216919,  -52.48002144,  -51.50661306, ...,  -52.15294131,
          -54.28691959,           -inf],
        [-170.16290562, -124.387798  ,  -95.8733683 , ..., -144.94463101,
         -109.47242906,           -inf],
        ...,
        [-400.66476654, -356.44800667, -382.54301712, ..., -343.55073065,
         -328.7593572 ,           -inf],
        [-370.71773152, -349.10819019, -353.2933865 , ..., -340.73893645,
         -340.99344747,           -inf],
        [-364.27365828, -340.86124326, -363.21636358, ..., -333.99655762,
         -381.75870365,           -inf]])
```

KSHITIJ GULATI
2017162

QUESTION 2:
In question 2, we generate the MFCC features of the audio file from scratch. We take the filter value as 40 and coeff value as 13 For example:

```
array([[-247.84409097,    5.65036365, -186.3950023 , ...,  279.29842711,
        -264.02809627,  233.45202509],
       [-249.47951549,    7.05137534, -186.4923291 , ...,  280.8518594 ,
        -265.68484251,  234.77755872],
       [-253.59145563,   12.59152087, -190.44002241, ...,  283.56405398,
        -270.23334144,  240.06560947],
       ...,
       [-251.0816892 ,    7.53812336, -186.16089298, ...,  282.54400112,
        -267.75809519,  236.52941216],
       [-246.39654578,    7.32020682, -184.0706272 , ...,  276.58678958,
        -264.30726843,  234.94325039],
       [   0.        ,    0.        ,    0.        , ...,    0.        ,
           0.        ,    0.        ]])
```

It is a more representative and sampled form of the short time fourier transform and is smaller in size than the spectrogram and subsequently has less features.

QUESTION 3:
First we test generate the mfcc and spectrogram features for all audio files(both training and validation) without any noise augmentation. Next, we train the models on the features generated from the training data and then test the efficiency of the model on the validation data. The following is the classification report:

```
accuracy for model_spec
0.5890136327185245
              precision    recall  f1-score   support

           0       0.69      0.71      0.70       252
           1       0.48      0.49      0.48       228
           2       0.35      0.53      0.42       155
           3       0.62      0.54      0.58       286
           4       0.74      0.59      0.66       351
           5       0.59      0.60      0.59       239
           6       0.75      0.62      0.68       315
           7       0.59      0.71      0.64       216
           8       0.56      0.50      0.53       274
           9       0.46      0.59      0.51       178

    accuracy                           0.59      2494
   macro avg       0.58      0.59      0.58      2494
weighted avg       0.61      0.59      0.59      2494

accuracy for model_mfcc
0.57457898957498
              precision    recall  f1-score   support

           0       0.67      0.75      0.71       231
           1       0.70      0.45      0.54       358
           2       0.28      0.44      0.34       147
           3       0.62      0.46      0.53       333
           4       0.70      0.72      0.71       272
           5       0.49      0.76      0.59       155
           6       0.43      0.89      0.58       127
           7       0.63      0.59      0.61       282
           8       0.68      0.44      0.54       377
           9       0.53      0.58      0.56       212

    accuracy                           0.57      2494
   macro avg       0.57      0.61      0.57      2494
weighted avg       0.61      0.57      0.57      2494
```

From this, we can see that the both models work at almost same accuracy, with spectrogram being slightly higher(59%) and mfcc(57%). The spectrogram model best predicts classes 0 and 6 whereas the mfcc model best predicts classes 0 and 4 based on high precision, recall and f1-scores. Both models are least effective when it comes to class 2 based on the same logic.

In this case, we first perform a random noise augmentation on the audio files before fitting the model. The classification report is as follows:

We can see that the MFCC model has a higher accuracy(43%) in comparison with that of spectrogram(38%) which tells us that the mfcc model is more robust as it performs better with noisy audio. The spectrogram model best classifies 6 and the mfcc model best classifies 0 based on higher f1 scores.

```
accuracy for model_spec with noise
0.3821170809943865
              precision    recall  f1-score   support

           0       0.12      0.88      0.20        34
           1       0.25      0.43      0.31       132
           2       0.25      0.30      0.27       201
           3       0.44      0.44      0.44       247
           4       0.42      0.60      0.49       194
           5       0.69      0.28      0.40       593
           6       0.39      0.77      0.52       133
           7       0.30      0.57      0.39       138
           8       0.82      0.27      0.40       743
           9       0.14      0.41      0.21        79

    accuracy                           0.38      2494
   macro avg       0.38      0.50      0.37      2494
weighted avg       0.56      0.38      0.40      2494

accuracy for model_mfcc with noise
0.4318364073777065
              precision    recall  f1-score   support

           0       0.41      0.71      0.52       150
           1       0.12      0.44      0.19        64
           2       0.29      0.35      0.31       196
           3       0.40      0.36      0.38       276
           4       0.37      0.59      0.46       176
           5       0.80      0.27      0.40       720
           6       0.58      0.60      0.59       254
           7       0.41      0.54      0.46       199
           8       0.52      0.48      0.50       261
           9       0.40      0.46      0.43       198

    accuracy                           0.43      2494
   macro avg       0.43      0.48      0.42      2494
weighted avg       0.53      0.43      0.43      2494
```