

```
In [1]: import pandas as pd
import numpy as np
import os
import seaborn as sns
import matplotlib.pyplot as plt
import librosa
import librosa.display
from IPython.display import Audio
import warnings
warnings.filterwarnings('ignore')
```

```
In [3]: paths = []
labels = []
for dirname, _, filenames in os.walk('Asthma Detection Dataset Version 2'):
    for filename in filenames:
        paths.append(os.path.join(dirname, filename))
        label = filename.split('_')[0]
        label = label.split('.')[0]
        labels.append(label.lower())
    if len(paths) == 1211:
        break
print('Dataset is Loaded')
```

Dataset is Loaded

```
In [4]: len(paths)
```

```
Out[4]: 1211
```

```
In [5]: paths[:5]
```

```
Out[5]: ['Asthma Detection Dataset Version 2\\asthma\\Asthma_RS_20 (1).wav',
'Asthma Detection Dataset Version 2\\asthma\\Asthma_RS_20 (10).wav',
'Asthma Detection Dataset Version 2\\asthma\\Asthma_RS_20 (100).wav',
'Asthma Detection Dataset Version 2\\asthma\\Asthma_RS_20 (101).wav',
'Asthma Detection Dataset Version 2\\asthma\\Asthma_RS_20 (102).wav']
```

```
In [6]: labels[:5]
```

```
Out[6]: ['asthma', 'asthma', 'asthma', 'asthma', 'asthma']
```

```
In [7]: ## Create a dataframe
df = pd.DataFrame()
df['sound'] = paths
df['label'] = labels
```

```
In [8]: df.head()
```

```
Out[8]:
```

	sound	label
0	Asthma Detection Dataset Version 2\asthma\Asth...	asthma
1	Asthma Detection Dataset Version 2\asthma\Asth...	asthma
2	Asthma Detection Dataset Version 2\asthma\Asth...	asthma
3	Asthma Detection Dataset Version 2\asthma\Asth...	asthma
4	Asthma Detection Dataset Version 2\asthma\Asth...	asthma

```
In [9]: df.tail()
```

```
Out[9]:
```

	sound	label
1206	Asthma Detection Dataset Version 2\pneumonia\P...	pneumonia
1207	Asthma Detection Dataset Version 2\pneumonia\P...	pneumonia
1208	Asthma Detection Dataset Version 2\pneumonia\P...	pneumonia
1209	Asthma Detection Dataset Version 2\pneumonia\P...	pneumonia
1210	Asthma Detection Dataset Version 2\pneumonia\P...	pneumonia

```
In [10]: df.sample(20)
```

Out[10]:

		sound	label
932	Asthma Detection Dataset Version 2\pneumonia\P...		pneumonia
482	Asthma Detection Dataset Version 2\copd\COPD_M...		copd
890	Asthma Detection Dataset Version 2\healthy\Hea...		healthy
1108	Asthma Detection Dataset Version 2\pneumonia\P...		pneumonia
825	Asthma Detection Dataset Version 2\healthy\Hea...		healthy
588	Asthma Detection Dataset Version 2\copd\COPD_M...		copd
502	Asthma Detection Dataset Version 2\copd\COPD_M...		copd
372	Asthma Detection Dataset Version 2\Bronchial\B...		bronchial
799	Asthma Detection Dataset Version 2\healthy\Hea...		healthy
652	Asthma Detection Dataset Version 2\copd\COPD_M...		copd
828	Asthma Detection Dataset Version 2\healthy\Hea...		healthy
317	Asthma Detection Dataset Version 2\Bronchial\B...		bronchial
820	Asthma Detection Dataset Version 2\healthy\Hea...		healthy
431	Asthma Detection Dataset Version 2\copd\COPD_M...		copd
246	Asthma Detection Dataset Version 2\asthma\Asth...		asthma
374	Asthma Detection Dataset Version 2\Bronchial\B...		bronchial
1099	Asthma Detection Dataset Version 2\pneumonia\P...		pneumonia
217	Asthma Detection Dataset Version 2\asthma\Asth...		asthma
34	Asthma Detection Dataset Version 2\asthma\Asth...		asthma
165	Asthma Detection Dataset Version 2\asthma\Asth...		asthma

In [11]: df.sample(30)

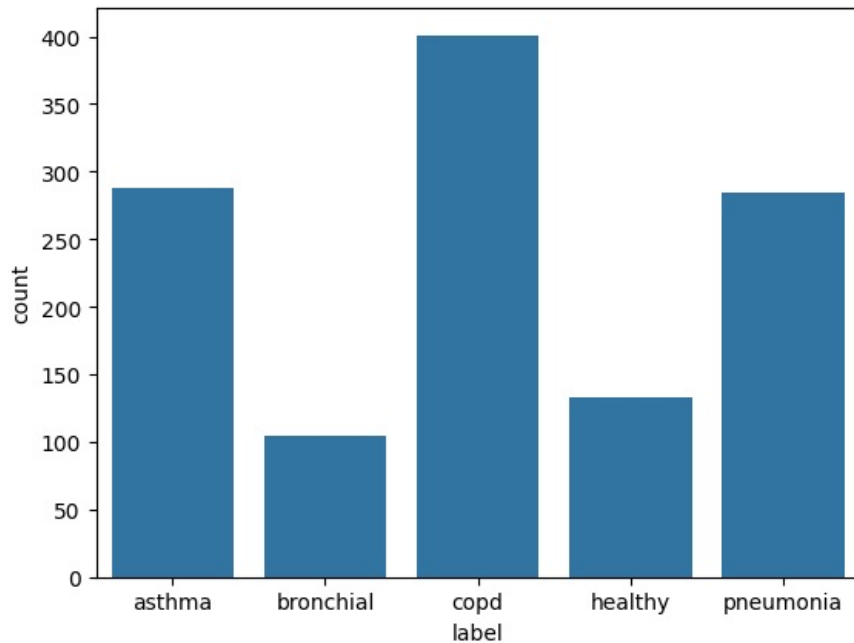
	sound	label
326	Asthma Detection Dataset Version 2\Bronchial\B...	bronchial
1155	Asthma Detection Dataset Version 2\pneumonia\P...	pneumonia
41	Asthma Detection Dataset Version 2\asthma\Asth...	asthma
504	Asthma Detection Dataset Version 2\copd\COPD_M...	copd
71	Asthma Detection Dataset Version 2\asthma\Asth...	asthma
29	Asthma Detection Dataset Version 2\asthma\Asth...	asthma
110	Asthma Detection Dataset Version 2\asthma\Asth...	asthma
1176	Asthma Detection Dataset Version 2\pneumonia\P...	pneumonia
949	Asthma Detection Dataset Version 2\pneumonia\P...	pneumonia
354	Asthma Detection Dataset Version 2\Bronchial\B...	bronchial
231	Asthma Detection Dataset Version 2\asthma\Asth...	asthma
630	Asthma Detection Dataset Version 2\copd\COPD_M...	copd
1143	Asthma Detection Dataset Version 2\pneumonia\P...	pneumonia
137	Asthma Detection Dataset Version 2\asthma\Asth...	asthma
305	Asthma Detection Dataset Version 2\Bronchial\B...	bronchial
525	Asthma Detection Dataset Version 2\copd\COPD_M...	copd
193	Asthma Detection Dataset Version 2\asthma\Asth...	asthma
985	Asthma Detection Dataset Version 2\pneumonia\P...	pneumonia
956	Asthma Detection Dataset Version 2\pneumonia\P...	pneumonia
764	Asthma Detection Dataset Version 2\copd\COPD_M...	copd
730	Asthma Detection Dataset Version 2\copd\COPD_M...	copd
1033	Asthma Detection Dataset Version 2\pneumonia\P...	pneumonia
1140	Asthma Detection Dataset Version 2\pneumonia\P...	pneumonia
734	Asthma Detection Dataset Version 2\copd\COPD_M...	copd
507	Asthma Detection Dataset Version 2\copd\COPD_M...	copd
1091	Asthma Detection Dataset Version 2\pneumonia\P...	pneumonia
859	Asthma Detection Dataset Version 2\healthy\Hea...	healthy
617	Asthma Detection Dataset Version 2\copd\COPD_M...	copd
535	Asthma Detection Dataset Version 2\copd\COPD_M...	copd
51	Asthma Detection Dataset Version 2\asthma\Asth...	asthma

```
In [12]: df['label'].value_counts()
```

```
Out[12]: label
copd      401
asthma    288
pneumonia 285
healthy   133
bronchial 104
Name: count, dtype: int64
```

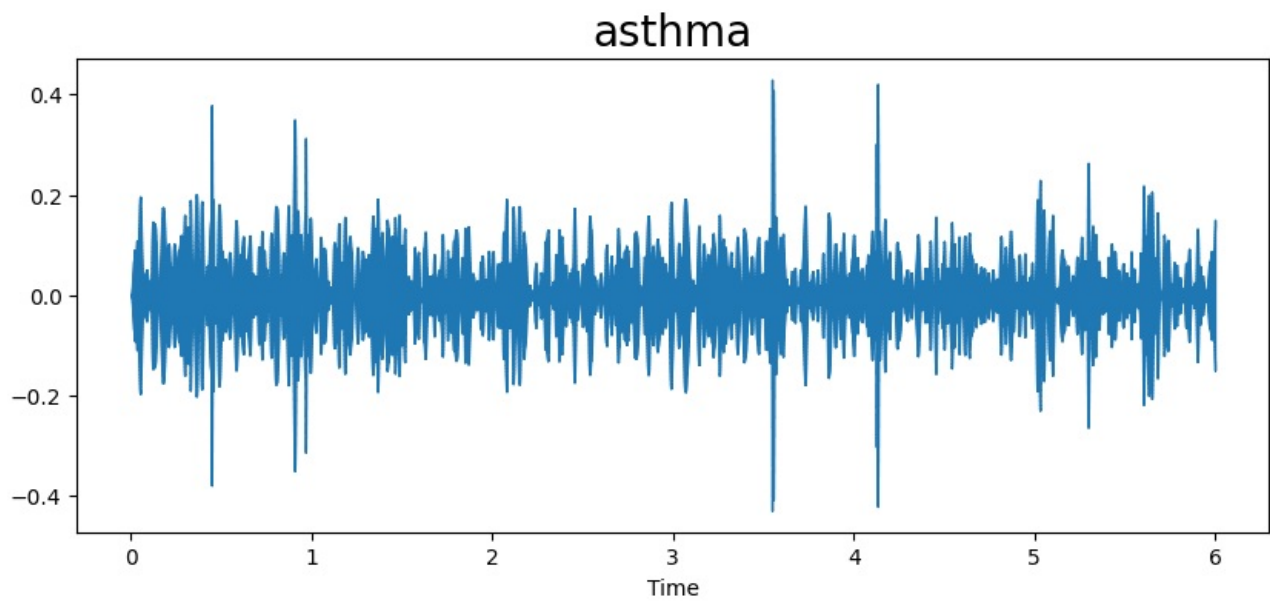
```
In [13]: sns.countplot(data=df, x='label')
```

```
Out[13]: <Axes: xlabel='label', ylabel='count'>
```

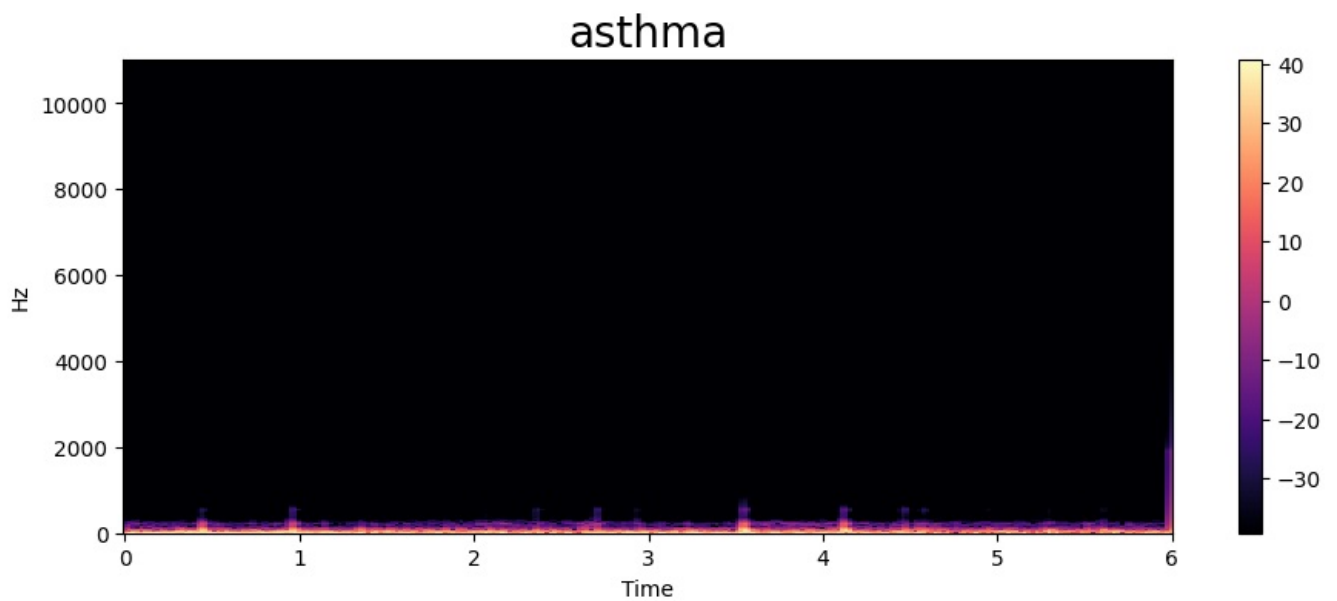


```
In [14]: def waveplot(data, sr, audio):  
    plt.figure(figsize=(10,4))  
    plt.title(audio, size=20)  
    librosa.display.waveshow(data, sr=sr)  
    plt.show()  
  
    def spectrogram(data, sr, audio):  
        x = librosa.stft(data)  
        xdb = librosa.amplitude_to_db(abs(x))  
        plt.figure(figsize=(11,4))  
        plt.title(audio, size=20)  
        librosa.display.specshow(xdb, sr=sr, x_axis='time', y_axis='hz')  
        plt.colorbar()
```

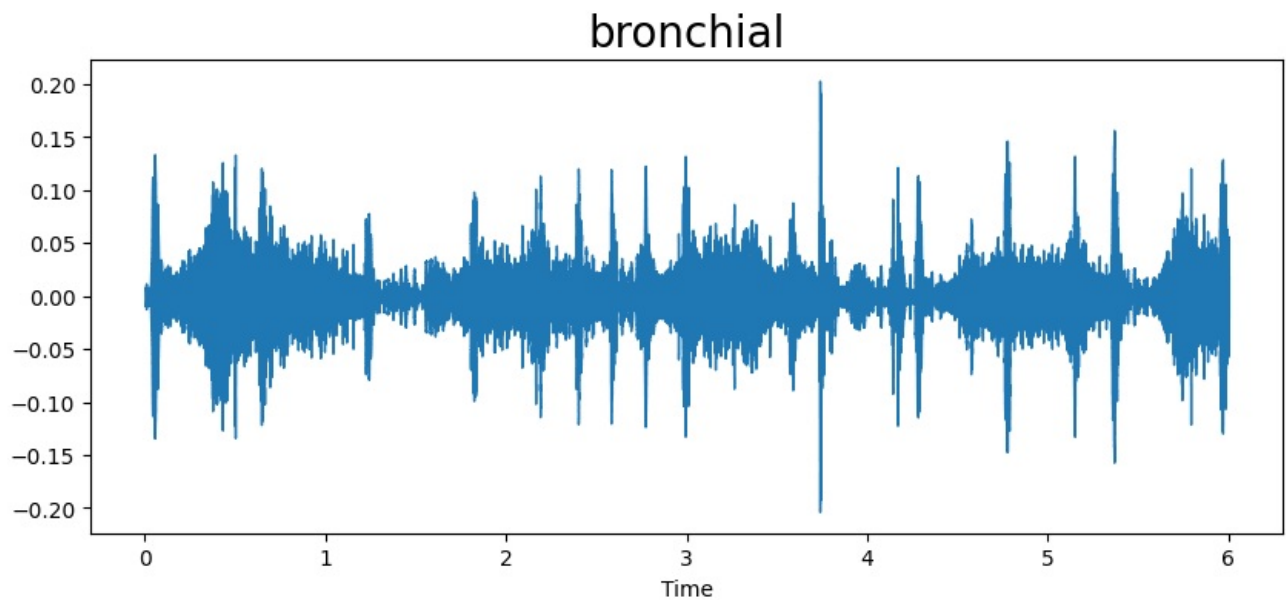
```
In [15]: audio = 'asthma'  
path = np.array(df['sound'][df['label']==audio])[0]  
data, sampling_rate = librosa.load(path)  
waveplot(data, sampling_rate, audio)  
spectrogram(data, sampling_rate, audio)  
Audio(path)
```



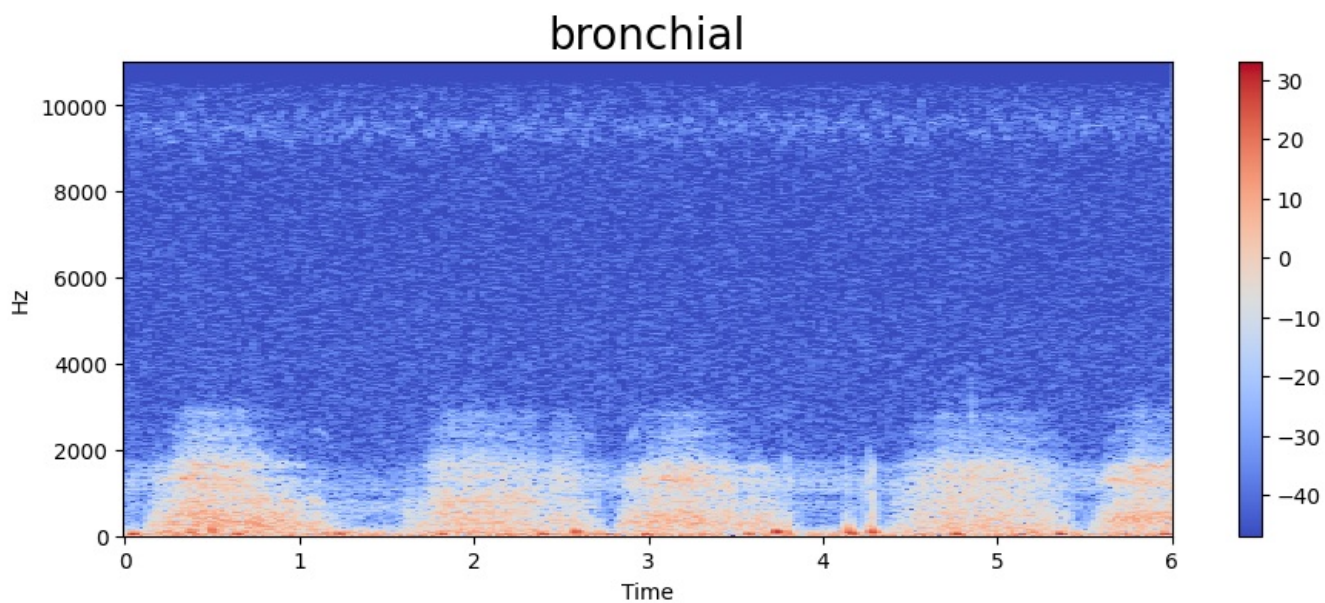
Out[15]: Your browser does not support the audio element.



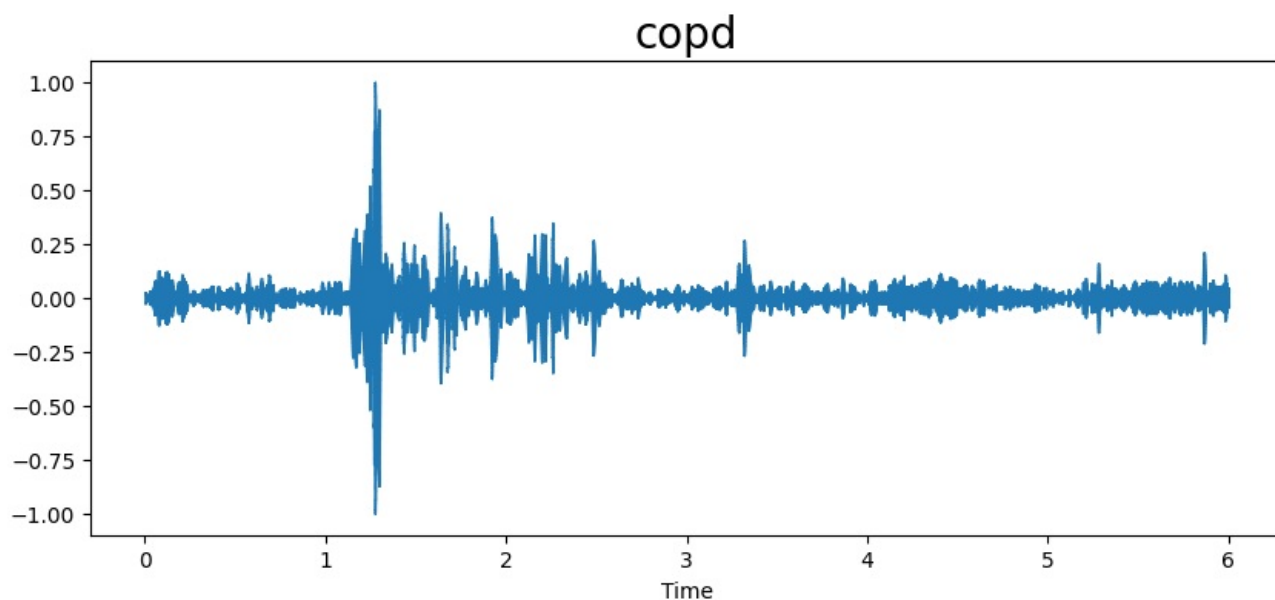
```
In [16]: audio = 'bronchial'
path = np.array(df['sound'])[df['label']==audio][0]
data, sampling_rate = librosa.load(path)
waveplot(data, sampling_rate, audio)
spectrogram(data, sampling_rate, audio)
Audio(path)
```



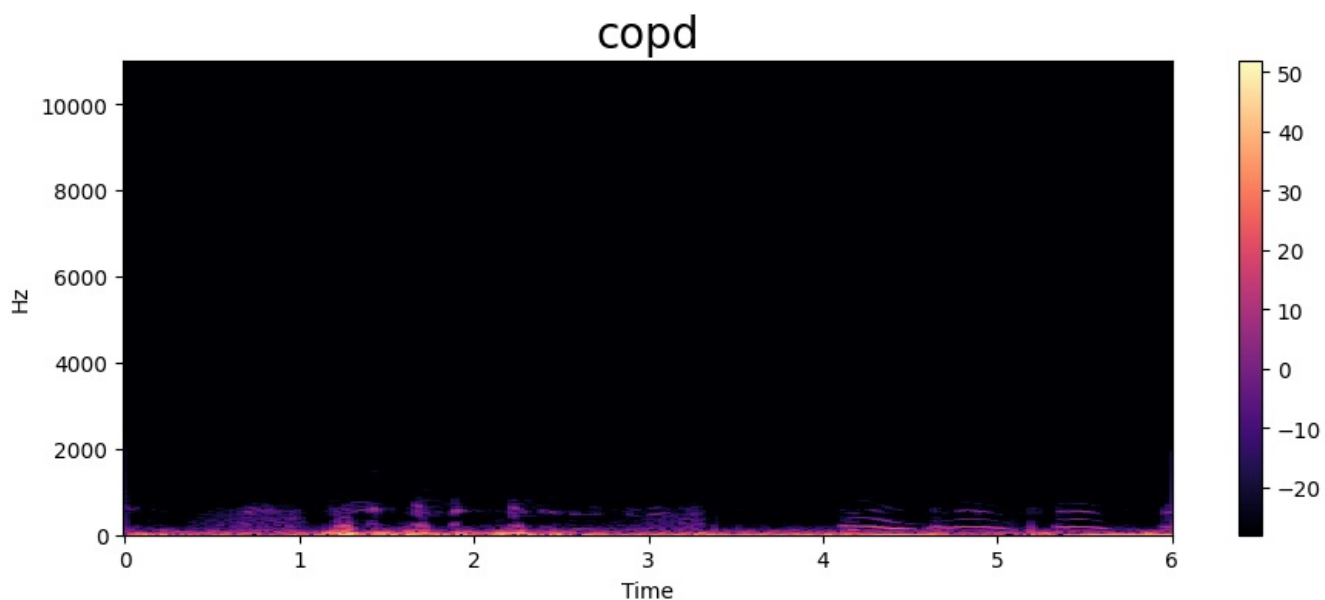
Out[16]: Your browser does not support the audio element.



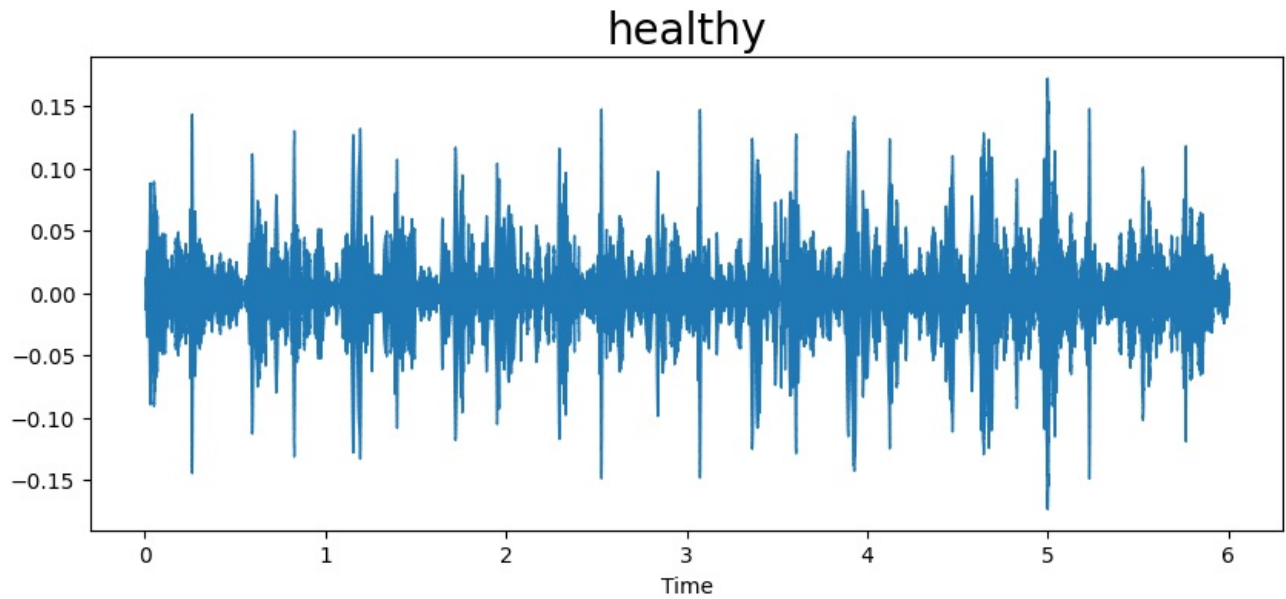
```
In [17]: audio = 'copd'
path = np.array(df['sound'][df['label']==audio])[0]
data, sampling_rate = librosa.load(path)
waveplot(data, sampling_rate, audio)
spectrogram(data, sampling_rate, audio)
Audio(path)
```



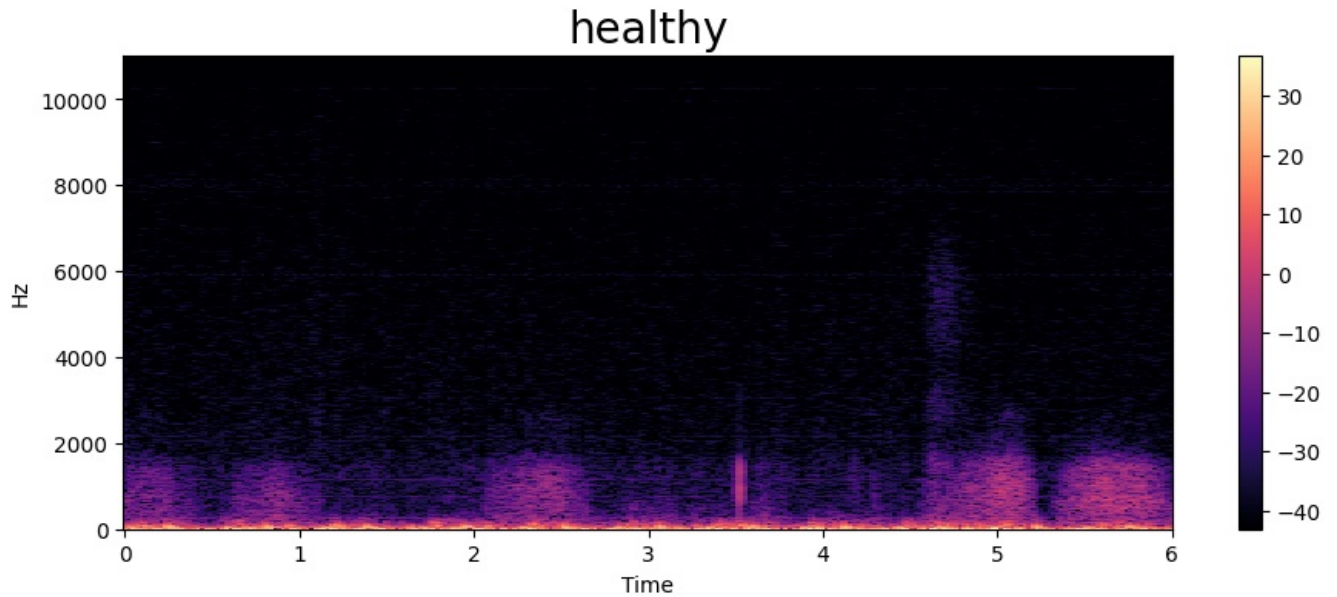
Out[17]: Your browser does not support the audio element.



```
In [18]: audio = 'healthy'
path = np.array(df['sound'][df['label']==audio])[0]
data, sampling_rate = librosa.load(path)
waveplot(data, sampling_rate, audio)
spectrogram(data, sampling_rate, audio)
Audio(path)
```

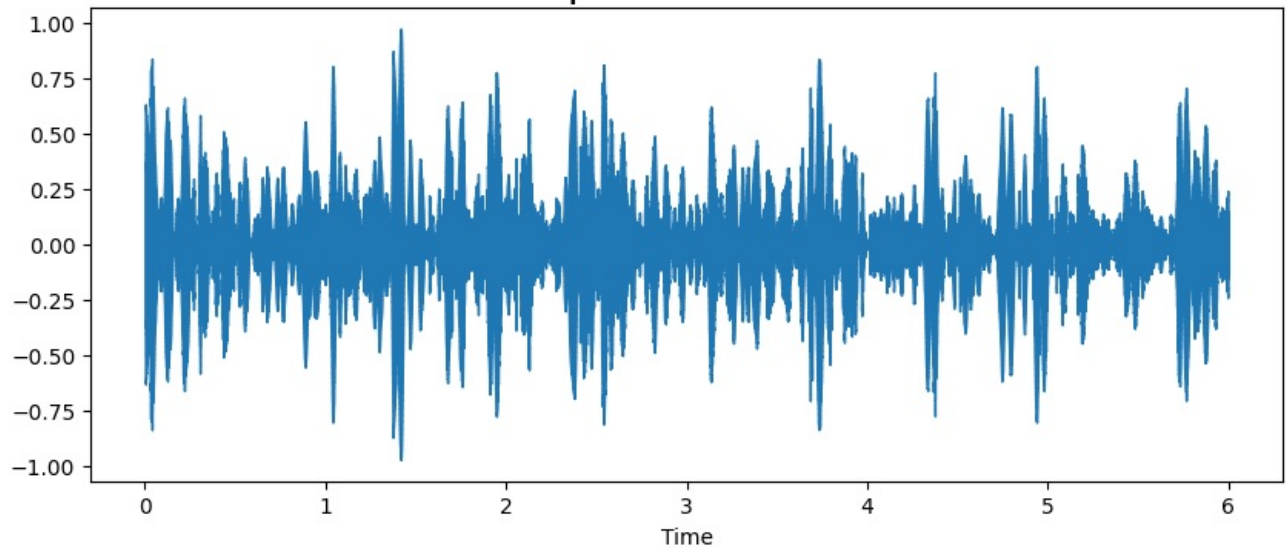


Out[18]: Your browser does not support the audio element.



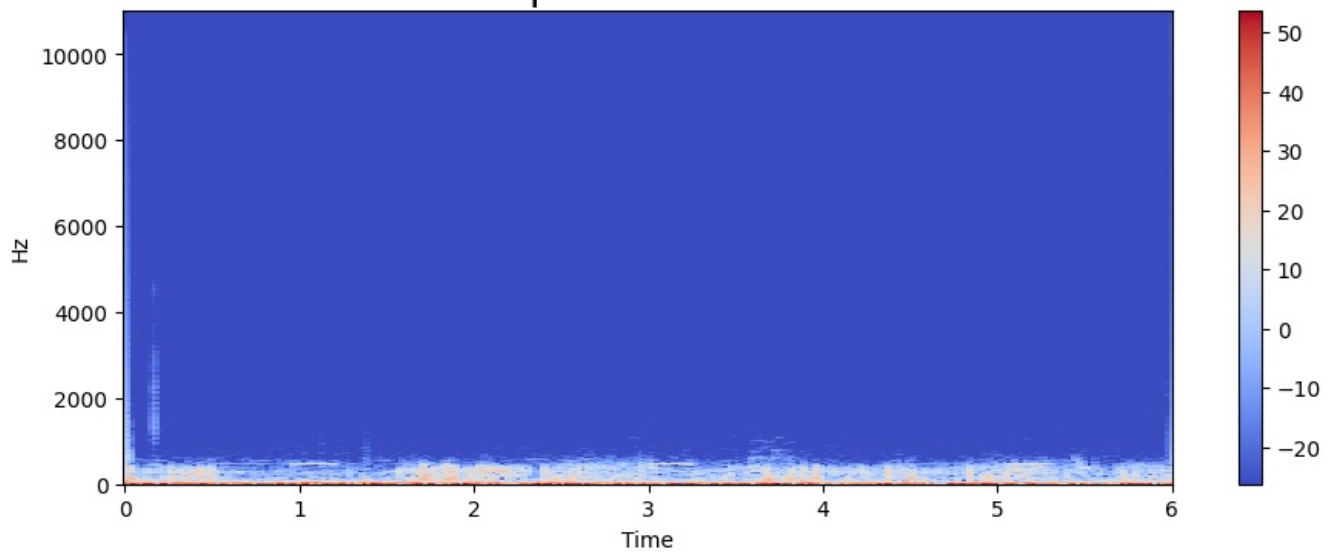
```
In [19]: audio = 'pneumonia'
path = np.array(df['sound'][df['label']==audio])[0]
data, sampling_rate = librosa.load(path)
waveplot(data, sampling_rate, audio)
spectrogram(data, sampling_rate, audio)
Audio(path)
```


pneumonia



Out[19]: Your browser does not support the audio element.

pneumonia



```
In [20]: def extract_mfcc(filename):
y, sr = librosa.load(filename, duration=3, offset=0.5)
mfcc = np.mean(librosa.feature.mfcc(y=y, sr=sr, n_mfcc=40).T, axis=0)
return mfcc
```

```
In [21]: extract_mfcc(df['sound'][0])
```

```
Out[21]: array([-548.83496 ,  50.119514 ,  45.78449 ,  43.384907 ,
  40.99531 ,  37.92376 ,  34.31658 ,  30.903397 ,
  27.728077 ,  24.546072 ,  21.37009 ,  18.457767 ,
  15.855426 ,  13.384048 ,  11.001728 ,  8.931421 ,
  7.295938 ,  5.979423 ,  4.9392314,  4.3285704,
  4.1767793,  4.285747 ,  4.5153503,  4.891299 ,
  5.3693304,  5.767992 ,  5.9868765,  6.0878544,
  6.109491 ,  5.9857993,  5.6918344,  5.316192 ,
  4.9219117,  4.4640946,  3.9046216,  3.3001502,
  2.7046251,  2.0866601,  1.4203374,  0.7722891],
      dtype=float32)
```

```
In [22]: X_mfcc = df['sound'].apply(lambda x: extract_mfcc(x))
```

```
In [23]: X_mfcc
```



```
Out[23]: 0      [-548.83496, 50.119514, 45.78449, 43.384907, 4...
1      [-510.55096, 64.57206, 58.473934, 54.521935, 5...
2      [-491.56403, 122.547005, 97.057144, 68.30173, ...
3      [-410.38647, 96.17459, 71.43918, 44.839233, 25...
4      [-584.2018, 149.43059, 109.23822, 64.08953, 26...
...
1206    [-472.8342, 57.394924, 52.172047, 48.309334, 4...
1207    [-514.3326, 67.60713, 61.357025, 55.430042, 48...
1208    [-408.67758, 100.07234, 81.82103, 62.381306, 4...
1209    [-451.33817, 125.4286, 52.841415, 16.206491, 1...
1210    [-385.91092, 113.33983, 90.27464, 67.36163, 44...
Name: sound, Length: 1211, dtype: object
```

```
In [24]: X = [x for x in X_mfcc]
X = np.array(X)
X.shape
```

```
Out[24]: (1211, 40)
```

```
In [25]: y=np.array(df['label'].tolist())
```

```
In [26]: y.shape
```

```
Out[26]: (1211,)
```

```
In [27]: X
```

```
Out[27]: array([[ -548.83496 ,  50.119514 ,  45.78449 , ...,  2.0866601,
                  1.4203374,  0.7722891],
                [ -510.55096 ,  64.57206 ,  58.473934 , ...,  2.840531 ,
                  2.3669758,  1.8308487],
                [ -491.56403 , 122.547005 ,  97.057144 , ...,  4.366303 ,
                  3.3435152,  2.3243194],
                ...,
                [ -408.67758 , 100.07234 ,  81.82103 , ...,  2.2515595,
                  2.4373362,  2.8444352],
                [ -451.33817 , 125.4286 ,  52.841415 , ...,  1.9024341,
                  2.301741 ,  1.5902761],
                [ -385.91092 , 113.33983 ,  90.27464 , ...,  1.5085611,
                  1.7577705,  2.1316829]], dtype=float32)
```

```
In [28]: y
```

```
Out[28]: array(['asthma', 'asthma', 'asthma', ..., 'pneumonia', 'pneumonia',
                'pneumonia'], dtype='<U9')
```

```
In [29]: from imblearn.over_sampling import SMOTE
smote = SMOTE(random_state=42)
X_resampled, y_resampled = smote.fit_resample(X, y)
```

```
In [30]: X_resampled.shape
```

```
Out[30]: (2005, 40)
```

```
In [31]: y_resampled.shape
```

```
Out[31]: (2005,)
```

```
In [32]: X_resampled
```

```
Out[32]: array([[ -5.48834961e+02,  5.01195145e+01,  4.57844887e+01, ...,
                  2.08666015e+00,  1.42033744e+00,  7.72289097e-01],
                [ -5.10550964e+02,  6.45720596e+01,  5.84739342e+01, ...,
                  2.84053111e+00,  2.36697578e+00,  1.83084869e+00],
                [ -4.91564026e+02,  1.22547005e+02,  9.70571442e+01, ...,
                  4.36630297e+00,  3.34351516e+00,  2.32431936e+00],
                ...,
                [ -4.79628448e+02,  1.23179596e+02,  5.41606674e+01, ...,
                  -4.48403448e-01,  2.29700789e-01,  6.30438566e-01],
                [ -4.96546539e+02,  1.02657547e+02,  5.94003410e+01, ...,
                  9.22891736e-01,  1.54411340e+00,  1.38727260e+00],
                [ -4.69754822e+02,  9.63708801e+01,  6.30505295e+01, ...,
                  4.35085803e-01,  7.85743356e-01,  8.84941578e-01]], dtype=float32)
```

```
In [33]: y_resampled
```

```
Out[33]: array(['asthma', 'asthma', 'asthma', ..., 'pneumonia', 'pneumonia',
                'pneumonia'], dtype='<U9')
```

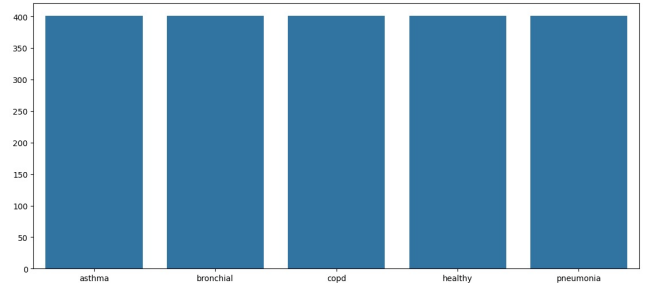
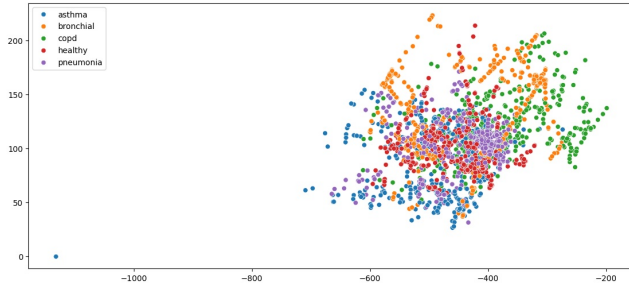
```
In [34]: np.unique(y, return_counts=True)
```

```
Out[34]: (array(['asthma', 'bronchial', 'copd', 'healthy', 'pneumonia'], dtype='<U9'),
array([288, 104, 401, 133, 285], dtype=int64))
```

```
In [35]: np.unique(y_resampled, return_counts=True)
```

```
Out[35]: (array(['asthma', 'bronchial', 'copd', 'healthy', 'pneumonia'], dtype='<U9'),
array([401, 401, 401, 401, 401], dtype=int64))
```

```
In [36]: fig, ax=plt.subplots(nrows=1, ncols=2, figsize=(30,6))
sns.scatterplot(x=X_resampled[:,0], y=X_resampled[:,1], hue=y_resampled, ax=ax[0])
idx, c=np.unique(y_resampled, return_counts=True)
sns.barplot(x=idx, y=c, ax=ax[1])
plt.show()
```



```
In [37]: from tensorflow.keras.utils import to_categorical
from sklearn.preprocessing import LabelEncoder
labelencoder=LabelEncoder()
y_resampled=to_categorical(labelencoder.fit_transform(y_resampled))
```

```
In [38]: y_resampled.shape
```

```
Out[38]: (2005, 5)
```

```
In [39]: X_resampled
```

```
Out[39]: array([[ -5.48834961e+02,  5.01195145e+01,  4.57844887e+01, ...,
        2.08666015e+00,  1.42033744e+00,  7.72289097e-01],
        [-5.10550964e+02,  6.45720596e+01,  5.84739342e+01, ...,
        2.84053111e+00,  2.36697578e+00,  1.83084869e+00],
        [-4.91564026e+02,  1.22547005e+02,  9.70571442e+01, ...,
        4.36630297e+00,  3.34351516e+00,  2.32431936e+00],
        ...,
        [-4.79628448e+02,  1.23179596e+02,  5.41606674e+01, ...,
        -4.48403448e-01,  2.29700789e-01,  6.30438566e-01],
        [-4.96546539e+02,  1.02657547e+02,  5.94003410e+01, ...,
        9.22891736e-01,  1.54411340e+00,  1.38727260e+00],
        [-4.69754822e+02,  9.63708801e+01,  6.30505295e+01, ...,
        4.35085803e-01,  7.85743356e-01,  8.84941578e-01]], dtype=float32)
```

```
In [40]: y_resampled
```

```
Out[40]: array([[1., 0., 0., 0., 0.],
        [1., 0., 0., 0., 0.],
        [1., 0., 0., 0., 0.],
        ...,
        [0., 0., 0., 0., 1.],
        [0., 0., 0., 0., 1.],
        [0., 0., 0., 0., 1.]])
```

```
In [41]: X_resampled.shape
```

```
Out[41]: (2005, 40)
```

```
In [42]: y_resampled.shape
```

```
Out[42]: (2005, 5)
```

```
In [43]: from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test=train_test_split(X_resampled,y_resampled,test_size=0.2,random_state=0)
```

```
In [44]: X_train
```

```
Out[44]: array([[ -5.75152100e+02,  6.93617935e+01,  5.48557549e+01, ...,
                1.88950646e+00,  1.69608390e+00,  1.20657635e+00],
               [-4.02211609e+02,  9.46986771e+01,  5.89261780e+01, ...,
                1.82643723e+00,  2.44722754e-01, -3.70863616e-01],
               [-4.77032867e+02,  1.20868042e+02,  5.64808655e+01, ...,
                6.10798039e-02,  4.33665454e-01,  2.71079212e-01],
               ...,
               [-4.84113190e+02,  1.14587006e+02,  4.15470352e+01, ...,
                1.10013235e+00,  3.49691749e-01,  1.18099129e+00],
               [-3.99894287e+02,  1.16806595e+02,  5.40612411e+01, ...,
                8.57568979e-01,  2.63239563e-01, -6.58950865e-01],
               [-4.40016235e+02,  1.02298325e+02,  6.68499146e+01, ...,
                2.66842890e+00,  1.67758262e+00,  6.89723551e-01]], dtype=float32)
```

```
In [45]: print(X_train.shape)
print(X_test.shape)
print(y_train.shape)
print(y_test.shape)
```

```
(1604, 40)
(401, 40)
(1604, 5)
(401, 5)
```

```
In [46]: import tensorflow as tf
```

```
In [47]: from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense,Activation,Dropout,BatchNormalization,Flatten
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.callbacks import ModelCheckpoint, EarlyStopping, ReduceLR0nPlateau
from datetime import datetime
from sklearn import metrics
```

```
In [54]: model=Sequential()

###first layer
model.add(Dense(100,input_shape=(40,)))
model.add(Activation('relu'))
model.add(Dropout(0.3))

###second layer
model.add(Dense(200))
model.add(Activation('relu'))
model.add(Dropout(0.3))

###third layer
model.add(Dense(100))
model.add(Activation('relu'))
model.add(Dropout(0.3))

###final layer
model.add(Dense(5))
model.add(Activation('softmax'))
```

```
In [55]: model.summary()
```

Model: "sequential_1"

Layer (type)	Output Shape	Param #
dense_4 (Dense)	(None, 100)	4,100
activation_4 (Activation)	(None, 100)	0
dropout_3 (Dropout)	(None, 100)	0
dense_5 (Dense)	(None, 200)	20,200
activation_5 (Activation)	(None, 200)	0
dropout_4 (Dropout)	(None, 200)	0
dense_6 (Dense)	(None, 100)	20,100
activation_6 (Activation)	(None, 100)	0
dropout_5 (Dropout)	(None, 100)	0
dense_7 (Dense)	(None, 5)	505
activation_7 (Activation)	(None, 5)	0

Total params: 44,905 (175.41 KB)

Trainable params: 44,905 (175.41 KB)

Non-trainable params: 0 (0.00 B)

```
In [56]: model.compile(loss='categorical_crossentropy', metrics=['accuracy'], optimizer='adam')
```

```
In [57]: ## Trianing my model
from tensorflow.keras.callbacks import ModelCheckpoint
from datetime import datetime

num_epochs = 150
num_batch_size = 32

checkpointer = ModelCheckpoint(filepath='saved_models/Asthma_audioclassification.keras',
                               verbose=1, save_best_only=True)

start = datetime.now()


history = model.fit(X_train, y_train, batch_size=num_batch_size, epochs=num_epochs, validation_data=(X_test, y_test))

duration = datetime.now() - start
print("Training time", duration)
```

Epoch 1/150

36/51  0s 1ms/step - accuracy: 0.1919 - loss: 33.8253


Epoch 1: val_loss improved from inf to 1.90047, saving model to saved_models/Asthma_audioclassification.keras

51/51  1s 5ms/step - accuracy: 0.1979 - loss: 29.9292 - val_accuracy: 0.2045 - val_loss: 1.9005

Epoch 2/150

34/51  0s 2ms/step - accuracy: 0.2280 - loss: 7.1544


Epoch 2: val_loss improved from 1.90047 to 1.57572, saving model to saved_models/Asthma_audioclassification.keras

51/51  0s 3ms/step - accuracy: 0.2259 - loss: 6.6593 - val_accuracy: 0.2594 - val_loss: 1.5757

Epoch 3/150

40/51  0s 1ms/step - accuracy: 0.2595 - loss: 2.9680


Epoch 3: val_loss did not improve from 1.57572

51/51  0s 2ms/step - accuracy: 0.2558 - loss: 2.9096 - val_accuracy: 0.1995 - val_loss: 1.6104

Epoch 4/150

36/51  0s 1ms/step - accuracy: 0.2016 - loss: 2.0635


Epoch 4: val_loss did not improve from 1.57572

51/51  0s 2ms/step - accuracy: 0.2081 - loss: 2.0532 - val_accuracy: 0.1970 - val_loss: 1.6114

Epoch 5/150

39/51  0s 1ms/step - accuracy: 0.2535 - loss: 1.7592


Epoch 5: val_loss did not improve from 1.57572

51/51  0s 2ms/step - accuracy: 0.2485 - loss: 1.7707 - val_accuracy: 0.2045 - val_loss: 1.6102

Epoch 6/150

33/51  0s 2ms/step - accuracy: 0.2104 - loss: 1.7303


Epoch 6: val_loss did not improve from 1.57572

51/51  0s 2ms/step - accuracy: 0.2168 - loss: 1.7351 - val_accuracy: 0.1895 - val_loss: 1.6117

Epoch 7/150

37/51  0s 1ms/step - accuracy: 0.2187 - loss: 1.7020


Epoch 7: val_loss did not improve from 1.57572

51/51  0s 2ms/step - accuracy: 0.2229 - loss: 1.6982 - val_accuracy: 0.1895 - val_loss: 1.6118

Epoch 8/150

47/51  0s 2ms/step - accuracy: 0.2486 - loss: 1.6435


Epoch 8: val_loss did not improve from 1.57572

51/51  0s 3ms/step - accuracy: 0.2482 - loss: 1.6437 - val_accuracy: 0.1895 - val_loss: 1.6118

Epoch 9/150

36/51  0s 1ms/step - accuracy: 0.2280 - loss: 1.6209


Epoch 9: val_loss did not improve from 1.57572

51/51  0s 2ms/step - accuracy: 0.2315 - loss: 1.6245 - val_accuracy: 0.1970 - val_loss: 1.6104

Epoch 10/150

35/51  0s 1ms/step - accuracy: 0.2241 - loss: 1.5912


Epoch 10: val_loss did not improve from 1.57572

51/51  0s 2ms/step - accuracy: 0.2292 - loss: 1.5958 - val_accuracy: 0.2195 - val_loss: 1.6028

Epoch 11/150





























37/51  0s 1ms/step - accuracy: 0.2510 - loss: 1.6315














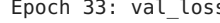

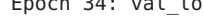




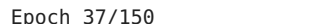

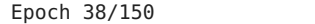

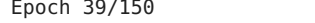




Epoch 11: val_loss improved from 1.57572 to 1.54338, saving model to saved_models/Asthma_audioclassification.keras






























51/51  0s 3ms/step - accuracy: 0.2510 - loss: 1.6224 - val_accuracy: 0.2843 - val_loss: 1.5434








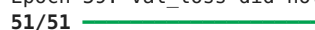


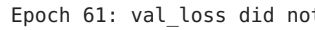




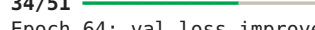

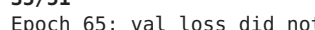




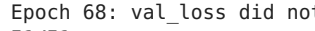





Epoch 12/150









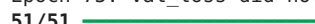





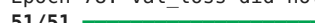

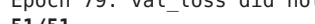
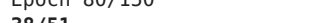
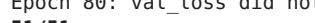
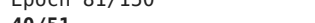
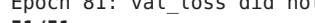
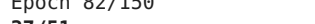
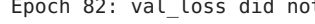
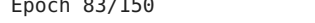
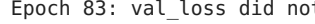
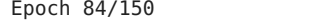
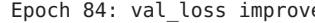





36/51  0s 1ms/step - accuracy: 0.2553 - loss: 1.5894









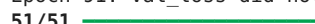

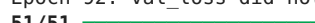

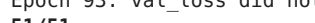
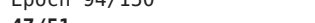
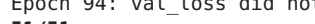
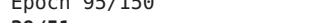

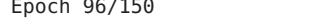

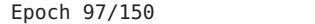

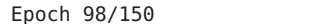

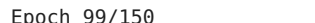








Epoch 12: val_loss did not improve from 1.54338
51/51  0s 2ms/step - accuracy: 0.2565 - loss: 1.5924 - val_accuracy: 0.2768 - val_loss: 1.5469
Epoch 13/150
37/51  0s 1ms/step - accuracy: 0.2779 - loss: 1.5540
Epoch 13: val_loss improved from 1.54338 to 1.53995, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 4ms/step - accuracy: 0.2777 - loss: 1.5581 - val_accuracy: 0.2868 - val_loss: 1.5400
Epoch 14/150
38/51  0s 1ms/step - accuracy: 0.2937 - loss: 1.5649
Epoch 14: val_loss improved from 1.53995 to 1.50121, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.2932 - loss: 1.5653 - val_accuracy: 0.3192 - val_loss: 1.5012
Epoch 15/150
37/51  0s 1ms/step - accuracy: 0.3054 - loss: 1.5310
Epoch 15: val_loss improved from 1.50121 to 1.45567, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.3047 - loss: 1.5318 - val_accuracy: 0.3516 - val_loss: 1.4557
Epoch 16/150
34/51  0s 2ms/step - accuracy: 0.3296 - loss: 1.4904
Epoch 16: val_loss improved from 1.45567 to 1.43787, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.3304 - loss: 1.4949 - val_accuracy: 0.3616 - val_loss: 1.4379
Epoch 17/150
32/51  0s 2ms/step - accuracy: 0.2881 - loss: 1.4970
Epoch 17: val_loss did not improve from 1.43787
51/51  0s 2ms/step - accuracy: 0.3007 - loss: 1.4893 - val_accuracy: 0.3566 - val_loss: 1.4408
Epoch 18/150
36/51  0s 1ms/step - accuracy: 0.3275 - loss: 1.4975
Epoch 18: val_loss improved from 1.43787 to 1.41187, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.3292 - loss: 1.4962 - val_accuracy: 0.3641 - val_loss: 1.4119
Epoch 19/150
30/51  0s 4ms/step - accuracy: 0.3426 - loss: 1.4598
Epoch 19: val_loss improved from 1.41187 to 1.35668, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 4ms/step - accuracy: 0.3491 - loss: 1.4493 - val_accuracy: 0.4314 - val_loss: 1.3567
Epoch 20/150
34/51  0s 2ms/step - accuracy: 0.3798 - loss: 1.4249
Epoch 20: val_loss improved from 1.35668 to 1.34865, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.3736 - loss: 1.4277 - val_accuracy: 0.4289 - val_loss: 1.3486
Epoch 21/150
37/51  0s 1ms/step - accuracy: 0.3855 - loss: 1.4279
Epoch 21: val_loss improved from 1.34865 to 1.25639, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.3852 - loss: 1.4213 - val_accuracy: 0.4663 - val_loss: 1.2564
Epoch 22/150
33/51  0s 2ms/step - accuracy: 0.3726 - loss: 1.3987
Epoch 22: val_loss improved from 1.25639 to 1.23032, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.3703 - loss: 1.3985 - val_accuracy: 0.4464 - val_loss: 1.2303
Epoch 23/150
34/51  0s 2ms/step - accuracy: 0.4124 - loss: 1.3430
Epoch 23: val_loss did not improve from 1.23032
51/51  0s 2ms/step - accuracy: 0.4071 - loss: 1.3457 - val_accuracy: 0.4863 - val_loss: 1.2371
Epoch 24/150
37/51  0s 1ms/step - accuracy: 0.4044 - loss: 1.3354
Epoch 24: val_loss improved from 1.23032 to 1.20787, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.4024 - loss: 1.3306 - val_accuracy: 0.4613 - val_loss: 1.2079
Epoch 25/150
36/51  0s 1ms/step - accuracy: 0.4002 - loss: 1.3454
Epoch 25: val_loss improved from 1.20787 to 1.19131, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.4060 - loss: 1.3346 - val_accuracy: 0.4589 - val_loss: 1.1913
Epoch 26/150
27/51  0s 2ms/step - accuracy: 0.4328 - loss: 1.2383
Epoch 26: val_loss improved from 1.19131 to 1.17584, saving model to saved_models/Asthma_audioclassification.keras













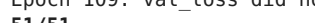
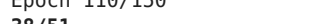
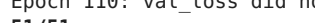
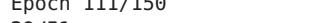

















51/51  0s 3ms/step - accuracy: 0.4300 - loss: 1.2512 - val_accuracy: 0.5112 - val_loss: 1.1758
Epoch 27/150
30/51  0s 2ms/step - accuracy: 0.4569 - loss: 1.2735
Epoch 27: val_loss improved from 1.17584 to 1.13784, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.4488 - loss: 1.2752 - val_accuracy: 0.5312 - val_loss: 1.1378
Epoch 28/150
35/51  0s 1ms/step - accuracy: 0.4806 - loss: 1.2675
Epoch 28: val_loss improved from 1.13784 to 1.12395, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.4772 - loss: 1.2651 - val_accuracy: 0.5461 - val_loss: 1.1240
Epoch 29/150
38/51  0s 3ms/step - accuracy: 0.4664 - loss: 1.2388
Epoch 29: val_loss improved from 1.12395 to 1.08826, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 4ms/step - accuracy: 0.4658 - loss: 1.2370 - val_accuracy: 0.5436 - val_loss: 1.0883
Epoch 30/150
39/51  0s 1ms/step - accuracy: 0.4941 - loss: 1.2220
Epoch 30: val_loss improved from 1.08826 to 1.04719, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.4945 - loss: 1.2091 - val_accuracy: 0.6010 - val_loss: 1.0472
Epoch 31/150
37/51  0s 1ms/step - accuracy: 0.4887 - loss: 1.1980
Epoch 31: val_loss improved from 1.04719 to 0.98053, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.4950 - loss: 1.1862 - val_accuracy: 0.6384 - val_loss: 0.9805
Epoch 32/150
43/51  0s 1ms/step - accuracy: 0.5403 - loss: 1.0857
Epoch 32: val_loss improved from 0.98053 to 0.92926, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 2ms/step - accuracy: 0.5428 - loss: 1.0859 - val_accuracy: 0.6733 - val_loss: 0.9293
Epoch 33/150
29/51  0s 4ms/step - accuracy: 0.5685 - loss: 1.0854
Epoch 33: val_loss improved from 0.92926 to 0.88784, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 4ms/step - accuracy: 0.5813 - loss: 1.0626 - val_accuracy: 0.7307 - val_loss: 0.8878
Epoch 34/150
27/51  0s 2ms/step - accuracy: 0.6350 - loss: 0.9883
Epoch 34: val_loss improved from 0.88784 to 0.81525, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.6268 - loss: 0.9948 - val_accuracy: 0.7631 - val_loss: 0.8153
Epoch 35/150
34/51  0s 2ms/step - accuracy: 0.6090 - loss: 0.9937
Epoch 35: val_loss improved from 0.81525 to 0.73927, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.6153 - loss: 0.9816 - val_accuracy: 0.7681 - val_loss: 0.7393
Epoch 36/150
42/51  0s 1ms/step - accuracy: 0.6365 - loss: 0.9046
Epoch 36: val_loss improved from 0.73927 to 0.72723, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.6370 - loss: 0.9105 - val_accuracy: 0.7756 - val_loss: 0.7272
Epoch 37/150
36/51  0s 1ms/step - accuracy: 0.6764 - loss: 0.8272
Epoch 37: val_loss improved from 0.72723 to 0.67774, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.6739 - loss: 0.8416 - val_accuracy: 0.7955 - val_loss: 0.6777
Epoch 38/150
41/51  0s 1ms/step - accuracy: 0.6873 - loss: 0.8505
Epoch 38: val_loss improved from 0.67774 to 0.62924, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 2ms/step - accuracy: 0.6894 - loss: 0.8483 - val_accuracy: 0.7805 - val_loss: 0.6292
Epoch 39/150
30/51  0s 2ms/step - accuracy: 0.6784 - loss: 0.8470
Epoch 39: val_loss did not improve from 0.62924
51/51  0s 2ms/step - accuracy: 0.6826 - loss: 0.8466 - val_accuracy: 0.8055 - val_loss: 0.6569
Epoch 40/150
35/51  0s 2ms/step - accuracy: 0.7189 - loss: 0.7647
Epoch 40: val_loss did not improve from 0.62924
51/51  0s 2ms/step - accuracy: 0.7173 - loss: 0.7733 - val_accuracy: 0.7805 - val_loss: 0.63




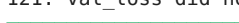

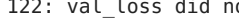












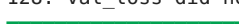

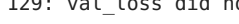

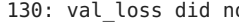

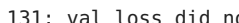

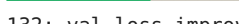




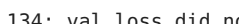
86
Epoch 41/150
42/51  0s 1ms/step - accuracy: 0.7245 - loss: 0.7611
Epoch 41: val_loss improved from 0.62924 to 0.61173, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.7259 - loss: 0.7609 - val_accuracy: 0.8005 - val_loss: 0.6117
Epoch 42/150
45/51  0s 1ms/step - accuracy: 0.7449 - loss: 0.7524
Epoch 42: val_loss improved from 0.61173 to 0.58295, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 2ms/step - accuracy: 0.7433 - loss: 0.7501 - val_accuracy: 0.8130 - val_loss: 0.5829
Epoch 43/150
31/51  0s 2ms/step - accuracy: 0.7424 - loss: 0.6936
Epoch 43: val_loss improved from 0.58295 to 0.58018, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.7410 - loss: 0.7067 - val_accuracy: 0.7980 - val_loss: 0.5802
Epoch 44/150
39/51  0s 1ms/step - accuracy: 0.7454 - loss: 0.6948
Epoch 44: val_loss did not improve from 0.58018
51/51  0s 3ms/step - accuracy: 0.7465 - loss: 0.6982 - val_accuracy: 0.7805 - val_loss: 0.6303
Epoch 45/150
38/51  0s 1ms/step - accuracy: 0.7476 - loss: 0.6838
Epoch 45: val_loss improved from 0.58018 to 0.55470, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.7463 - loss: 0.6894 - val_accuracy: 0.8180 - val_loss: 0.5547
Epoch 46/150
28/51  0s 2ms/step - accuracy: 0.7671 - loss: 0.6079
Epoch 46: val_loss did not improve from 0.55470
51/51  0s 2ms/step - accuracy: 0.7559 - loss: 0.6367 - val_accuracy: 0.8030 - val_loss: 0.5712
Epoch 47/150
40/51  0s 3ms/step - accuracy: 0.7527 - loss: 0.6890
Epoch 47: val_loss did not improve from 0.55470
51/51  0s 4ms/step - accuracy: 0.7557 - loss: 0.6832 - val_accuracy: 0.8204 - val_loss: 0.5560
Epoch 48/150
36/51  0s 1ms/step - accuracy: 0.7479 - loss: 0.6640
Epoch 48: val_loss improved from 0.55470 to 0.53316, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.7497 - loss: 0.6576 - val_accuracy: 0.8030 - val_loss: 0.5332
Epoch 49/150
33/51  0s 2ms/step - accuracy: 0.7161 - loss: 0.7330
Epoch 49: val_loss improved from 0.53316 to 0.51362, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.7343 - loss: 0.6966 - val_accuracy: 0.8229 - val_loss: 0.5136
Epoch 50/150
37/51  0s 3ms/step - accuracy: 0.7803 - loss: 0.6262
Epoch 50: val_loss did not improve from 0.51362
51/51  0s 3ms/step - accuracy: 0.7801 - loss: 0.6250 - val_accuracy: 0.8329 - val_loss: 0.5217
Epoch 51/150
36/51  0s 1ms/step - accuracy: 0.7704 - loss: 0.5902
Epoch 51: val_loss improved from 0.51362 to 0.50181, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.7698 - loss: 0.5968 - val_accuracy: 0.8229 - val_loss: 0.5018
Epoch 52/150
30/51  0s 2ms/step - accuracy: 0.7707 - loss: 0.6720
Epoch 52: val_loss did not improve from 0.50181
51/51  0s 2ms/step - accuracy: 0.7721 - loss: 0.6606 - val_accuracy: 0.8254 - val_loss: 0.5044
Epoch 53/150
40/51  0s 1ms/step - accuracy: 0.7915 - loss: 0.5850
Epoch 53: val_loss improved from 0.50181 to 0.49019, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.7904 - loss: 0.5873 - val_accuracy: 0.8304 - val_loss: 0.4902
Epoch 54/150
35/51  0s 1ms/step - accuracy: 0.7705 - loss: 0.6323
Epoch 54: val_loss improved from 0.49019 to 0.48464, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.7733 - loss: 0.6237 - val_accuracy: 0.8229 - val_loss: 0.4846
Epoch 55/150
40/51  0s 3ms/step - accuracy: 0.7745 - loss: 0.5730
Epoch 55: val_loss did not improve from 0.48464






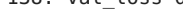
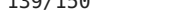
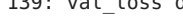







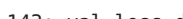










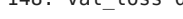



51/51  0s 3ms/step - accuracy: 0.7766 - loss: 0.5767 - val_accuracy: 0.8229 - val_loss: 0.4896
Epoch 56/150
35/51  0s 1ms/step - accuracy: 0.7827 - loss: 0.5989
Epoch 56: val_loss improved from 0.48464 to 0.45922, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.7863 - loss: 0.5844 - val_accuracy: 0.8429 - val_loss: 0.4592
Epoch 57/150
32/51  0s 2ms/step - accuracy: 0.7702 - loss: 0.5618
Epoch 57: val_loss improved from 0.45922 to 0.45917, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 4ms/step - accuracy: 0.7699 - loss: 0.5757 - val_accuracy: 0.8404 - val_loss: 0.4592
Epoch 58/150
34/51  0s 2ms/step - accuracy: 0.7973 - loss: 0.5737
Epoch 58: val_loss did not improve from 0.45917
51/51  0s 2ms/step - accuracy: 0.7972 - loss: 0.5686 - val_accuracy: 0.8454 - val_loss: 0.4841
Epoch 59/150
34/51  0s 2ms/step - accuracy: 0.7944 - loss: 0.5688
Epoch 59: val_loss did not improve from 0.45917
51/51  0s 2ms/step - accuracy: 0.7941 - loss: 0.5694 - val_accuracy: 0.8229 - val_loss: 0.4653
Epoch 60/150
41/51  0s 3ms/step - accuracy: 0.7844 - loss: 0.5714
Epoch 60: val_loss did not improve from 0.45917
51/51  0s 4ms/step - accuracy: 0.7879 - loss: 0.5652 - val_accuracy: 0.8579 - val_loss: 0.4724
Epoch 61/150
37/51  0s 1ms/step - accuracy: 0.7807 - loss: 0.5584
Epoch 61: val_loss did not improve from 0.45917
51/51  0s 2ms/step - accuracy: 0.7841 - loss: 0.5547 - val_accuracy: 0.8329 - val_loss: 0.4715
Epoch 62/150
35/51  0s 2ms/step - accuracy: 0.7808 - loss: 0.5710
Epoch 62: val_loss improved from 0.45917 to 0.42703, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.7854 - loss: 0.5645 - val_accuracy: 0.8554 - val_loss: 0.4270
Epoch 63/150
30/51  0s 3ms/step - accuracy: 0.7816 - loss: 0.5696
Epoch 63: val_loss did not improve from 0.42703
51/51  0s 3ms/step - accuracy: 0.7903 - loss: 0.5512 - val_accuracy: 0.8404 - val_loss: 0.4369
Epoch 64/150
34/51  0s 2ms/step - accuracy: 0.7895 - loss: 0.5514
Epoch 64: val_loss improved from 0.42703 to 0.42140, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.7976 - loss: 0.5365 - val_accuracy: 0.8504 - val_loss: 0.4214
Epoch 65/150
35/51  0s 2ms/step - accuracy: 0.8622 - loss: 0.4169
Epoch 65: val_loss did not improve from 0.42140
51/51  0s 3ms/step - accuracy: 0.8520 - loss: 0.4380 - val_accuracy: 0.8678 - val_loss: 0.4405
Epoch 66/150
35/51  0s 1ms/step - accuracy: 0.8035 - loss: 0.5364
Epoch 66: val_loss did not improve from 0.42140
51/51  0s 2ms/step - accuracy: 0.8073 - loss: 0.5300 - val_accuracy: 0.8504 - val_loss: 0.4764
Epoch 67/150
36/51  0s 1ms/step - accuracy: 0.8190 - loss: 0.4781
Epoch 67: val_loss did not improve from 0.42140
51/51  0s 2ms/step - accuracy: 0.8213 - loss: 0.4751 - val_accuracy: 0.8678 - val_loss: 0.4480
Epoch 68/150
32/51  0s 3ms/step - accuracy: 0.8395 - loss: 0.4856
Epoch 68: val_loss did not improve from 0.42140
51/51  0s 3ms/step - accuracy: 0.8268 - loss: 0.5005 - val_accuracy: 0.8529 - val_loss: 0.4423
Epoch 69/150
37/51  0s 1ms/step - accuracy: 0.8546 - loss: 0.4383
Epoch 69: val_loss did not improve from 0.42140
51/51 0s 2ms/step - accuracy: 0.8495 - loss: 0.4464 - val_accuracy: 0.8728 - val_loss: 0.4250
Epoch 70/150
39/51 0s 1ms/step - accuracy: 0.8156 - loss: 0.4841
Epoch 70: val_loss did not improve from 0.42140
51/51 0s 3ms/step - accuracy: 0.8138 - loss: 0.4871 - val_accuracy: 0.8628 - val_loss: 0.4446
Epoch 71/150
29/51 0s 2ms/step - accuracy: 0.8361 - loss: 0.4683

Epoch 71: val_loss did not improve from 0.42140
51/51  0s 2ms/step - accuracy: 0.8330 - loss: 0.4797 - val_accuracy: 0.8304 - val_loss: 0.4646
Epoch 72/150
39/51  0s 1ms/step - accuracy: 0.8201 - loss: 0.4616
Epoch 72: val_loss did not improve from 0.42140
51/51  0s 2ms/step - accuracy: 0.8205 - loss: 0.4641 - val_accuracy: 0.8678 - val_loss: 0.4300
Epoch 73/150
25/51  0s 2ms/step - accuracy: 0.8415 - loss: 0.4466
Epoch 73: val_loss did not improve from 0.42140
51/51  0s 2ms/step - accuracy: 0.8349 - loss: 0.4520 - val_accuracy: 0.8703 - val_loss: 0.4327
Epoch 74/150
37/51  0s 1ms/step - accuracy: 0.8225 - loss: 0.4778
Epoch 74: val_loss did not improve from 0.42140
51/51  0s 2ms/step - accuracy: 0.8284 - loss: 0.4662 - val_accuracy: 0.8728 - val_loss: 0.4258
Epoch 75/150
49/51  0s 3ms/step - accuracy: 0.8252 - loss: 0.4454
Epoch 75: val_loss did not improve from 0.42140
51/51  0s 3ms/step - accuracy: 0.8249 - loss: 0.4463 - val_accuracy: 0.8853 - val_loss: 0.4283
Epoch 76/150
38/51  0s 1ms/step - accuracy: 0.8232 - loss: 0.4792
Epoch 76: val_loss improved from 0.42140 to 0.39945, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.8265 - loss: 0.4770 - val_accuracy: 0.8828 - val_loss: 0.3994
Epoch 77/150
45/51  0s 3ms/step - accuracy: 0.8575 - loss: 0.4283
Epoch 77: val_loss improved from 0.39945 to 0.37900, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 4ms/step - accuracy: 0.8547 - loss: 0.4325 - val_accuracy: 0.8703 - val_loss: 0.3790
Epoch 78/150
40/51  0s 1ms/step - accuracy: 0.8286 - loss: 0.4660
Epoch 78: val_loss did not improve from 0.37900
51/51  0s 2ms/step - accuracy: 0.8273 - loss: 0.4674 - val_accuracy: 0.8678 - val_loss: 0.4023
Epoch 79/150
40/51  0s 3ms/step - accuracy: 0.8465 - loss: 0.4173
Epoch 79: val_loss did not improve from 0.37900
51/51  0s 3ms/step - accuracy: 0.8445 - loss: 0.4214 - val_accuracy: 0.8778 - val_loss: 0.3792
Epoch 80/150
38/51  0s 1ms/step - accuracy: 0.8544 - loss: 0.4005
Epoch 80: val_loss did not improve from 0.37900
51/51  0s 2ms/step - accuracy: 0.8509 - loss: 0.4128 - val_accuracy: 0.8678 - val_loss: 0.4199
Epoch 81/150
40/51  0s 1ms/step - accuracy: 0.8440 - loss: 0.4376
Epoch 81: val_loss did not improve from 0.37900
51/51  0s 3ms/step - accuracy: 0.8433 - loss: 0.4371 - val_accuracy: 0.8579 - val_loss: 0.4249
Epoch 82/150
37/51  0s 1ms/step - accuracy: 0.8117 - loss: 0.5421
Epoch 82: val_loss did not improve from 0.37900
51/51  0s 2ms/step - accuracy: 0.8162 - loss: 0.5275 - val_accuracy: 0.8678 - val_loss: 0.3903
Epoch 83/150
29/51  0s 2ms/step - accuracy: 0.8454 - loss: 0.3988
Epoch 83: val_loss did not improve from 0.37900
51/51  0s 3ms/step - accuracy: 0.8437 - loss: 0.4117 - val_accuracy: 0.8678 - val_loss: 0.4259
Epoch 84/150
36/51  0s 1ms/step - accuracy: 0.8397 - loss: 0.4738
Epoch 84: val_loss improved from 0.37900 to 0.36641, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.8417 - loss: 0.4606 - val_accuracy: 0.8728 - val_loss: 0.3664
Epoch 85/150
42/51  0s 1ms/step - accuracy: 0.8420 - loss: 0.4718
Epoch 85: val_loss did not improve from 0.36641
51/51  0s 2ms/step - accuracy: 0.8418 - loss: 0.4673 - val_accuracy: 0.8728 - val_loss: 0.4051
Epoch 86/150
34/51  0s 3ms/step - accuracy: 0.8219 - loss: 0.4897
Epoch 86: val_loss did not improve from 0.36641
51/51  0s 4ms/step - accuracy: 0.8256 - loss: 0.4724 - val_accuracy: 0.8653 - val_loss: 0.3846
Epoch 87/150
36/51  0s 1ms/step - accuracy: 0.8493 - loss: 0.4500

Epoch 87: val_loss did not improve from 0.36641
51/51  0s 2ms/step - accuracy: 0.8471 - loss: 0.4456 - val_accuracy: 0.8678 - val_loss: 0.3951
Epoch 88/150
27/51  0s 2ms/step - accuracy: 0.8411 - loss: 0.3807
Epoch 88: val_loss did not improve from 0.36641
51/51  0s 3ms/step - accuracy: 0.8418 - loss: 0.3927 - val_accuracy: 0.8603 - val_loss: 0.4152
Epoch 89/150
41/51  0s 1ms/step - accuracy: 0.8249 - loss: 0.5041
Epoch 89: val_loss did not improve from 0.36641
51/51  0s 2ms/step - accuracy: 0.8249 - loss: 0.5024 - val_accuracy: 0.8753 - val_loss: 0.4008
Epoch 90/150
43/51  0s 1ms/step - accuracy: 0.8550 - loss: 0.4381
Epoch 90: val_loss did not improve from 0.36641
51/51  0s 2ms/step - accuracy: 0.8514 - loss: 0.4424 - val_accuracy: 0.8628 - val_loss: 0.4187
Epoch 91/150
34/51  0s 2ms/step - accuracy: 0.8257 - loss: 0.4922
Epoch 91: val_loss did not improve from 0.36641
51/51  0s 2ms/step - accuracy: 0.8274 - loss: 0.4720 - val_accuracy: 0.8728 - val_loss: 0.3787
Epoch 92/150
33/51  0s 2ms/step - accuracy: 0.8263 - loss: 0.4589
Epoch 92: val_loss did not improve from 0.36641
51/51  0s 2ms/step - accuracy: 0.8348 - loss: 0.4390 - val_accuracy: 0.8703 - val_loss: 0.3689
Epoch 93/150
41/51  0s 1ms/step - accuracy: 0.8495 - loss: 0.4298
Epoch 93: val_loss did not improve from 0.36641
51/51  0s 2ms/step - accuracy: 0.8504 - loss: 0.4262 - val_accuracy: 0.8753 - val_loss: 0.3955
Epoch 94/150
47/51  0s 2ms/step - accuracy: 0.8352 - loss: 0.4406
Epoch 94: val_loss did not improve from 0.36641
51/51  0s 3ms/step - accuracy: 0.8350 - loss: 0.4401 - val_accuracy: 0.8678 - val_loss: 0.4057
Epoch 95/150
39/51  0s 1ms/step - accuracy: 0.8485 - loss: 0.4411
Epoch 95: val_loss did not improve from 0.36641
51/51  0s 2ms/step - accuracy: 0.8469 - loss: 0.4417 - val_accuracy: 0.8703 - val_loss: 0.3831
Epoch 96/150
37/51  0s 1ms/step - accuracy: 0.8548 - loss: 0.4014
Epoch 96: val_loss improved from 0.36641 to 0.34974, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.8537 - loss: 0.4032 - val_accuracy: 0.8853 - val_loss: 0.3497
Epoch 97/150
33/51  0s 3ms/step - accuracy: 0.8520 - loss: 0.3690
Epoch 97: val_loss did not improve from 0.34974
51/51  0s 4ms/step - accuracy: 0.8481 - loss: 0.3863 - val_accuracy: 0.8828 - val_loss: 0.3757
Epoch 98/150
38/51  0s 1ms/step - accuracy: 0.8441 - loss: 0.4189
Epoch 98: val_loss did not improve from 0.34974
51/51  0s 2ms/step - accuracy: 0.8486 - loss: 0.4066 - val_accuracy: 0.8678 - val_loss: 0.3813
Epoch 99/150
46/51  0s 1ms/step - accuracy: 0.8648 - loss: 0.3707
Epoch 99: val_loss did not improve from 0.34974
51/51  0s 2ms/step - accuracy: 0.8634 - loss: 0.3758 - val_accuracy: 0.8778 - val_loss: 0.4128
Epoch 100/150
33/51  0s 2ms/step - accuracy: 0.8448 - loss: 0.4287
Epoch 100: val_loss did not improve from 0.34974
51/51  0s 3ms/step - accuracy: 0.8470 - loss: 0.4206 - val_accuracy: 0.8703 - val_loss: 0.3773
Epoch 101/150
39/51  0s 1ms/step - accuracy: 0.8380 - loss: 0.4025
Epoch 101: val_loss improved from 0.34974 to 0.33734, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.8384 - loss: 0.3996 - val_accuracy: 0.8828 - val_loss: 0.3373
Epoch 102/150
37/51  0s 1ms/step - accuracy: 0.8744 - loss: 0.3333
Epoch 102: val_loss did not improve from 0.33734
51/51  0s 2ms/step - accuracy: 0.8697 - loss: 0.3420 - val_accuracy: 0.8703 - val_loss: 0.3885
Epoch 103/150
26/51  0s 2ms/step - accuracy: 0.8282 - loss: 0.4718
Epoch 103: val_loss did not improve from 0.33734

51/51  0s 2ms/step - accuracy: 0.8296 - loss: 0.4599 - val_accuracy: 0.8579 - val_loss: 0.3745
Epoch 104/150
42/51  0s 1ms/step - accuracy: 0.8515 - loss: 0.4220
Epoch 104: val_loss did not improve from 0.33734
51/51  0s 2ms/step - accuracy: 0.8471 - loss: 0.4265 - val_accuracy: 0.8828 - val_loss: 0.3732
Epoch 105/150
35/51  0s 1ms/step - accuracy: 0.8304 - loss: 0.4479
Epoch 105: val_loss did not improve from 0.33734
51/51  0s 3ms/step - accuracy: 0.8335 - loss: 0.4398 - val_accuracy: 0.8554 - val_loss: 0.3769
Epoch 106/150
42/51  0s 1ms/step - accuracy: 0.8517 - loss: 0.3906
Epoch 106: val_loss did not improve from 0.33734
51/51  0s 3ms/step - accuracy: 0.8502 - loss: 0.3931 - val_accuracy: 0.8878 - val_loss: 0.3825
Epoch 107/150
45/51  0s 1ms/step - accuracy: 0.8325 - loss: 0.4195
Epoch 107: val_loss did not improve from 0.33734
51/51  0s 2ms/step - accuracy: 0.8339 - loss: 0.4188 - val_accuracy: 0.8903 - val_loss: 0.3433
Epoch 108/150
39/51  0s 1ms/step - accuracy: 0.8593 - loss: 0.3645
Epoch 108: val_loss did not improve from 0.33734
51/51  0s 2ms/step - accuracy: 0.8581 - loss: 0.3672 - val_accuracy: 0.8753 - val_loss: 0.4013
Epoch 109/150
42/51  0s 3ms/step - accuracy: 0.8706 - loss: 0.3584
Epoch 109: val_loss did not improve from 0.33734
51/51  0s 3ms/step - accuracy: 0.8678 - loss: 0.3668 - val_accuracy: 0.8753 - val_loss: 0.3727
Epoch 110/150
38/51  0s 1ms/step - accuracy: 0.8543 - loss: 0.3783
Epoch 110: val_loss did not improve from 0.33734
51/51  0s 2ms/step - accuracy: 0.8520 - loss: 0.3892 - val_accuracy: 0.8728 - val_loss: 0.3804
Epoch 111/150
29/51  0s 2ms/step - accuracy: 0.8406 - loss: 0.4490
Epoch 111: val_loss did not improve from 0.33734
51/51  0s 3ms/step - accuracy: 0.8472 - loss: 0.4254 - val_accuracy: 0.8853 - val_loss: 0.3508
Epoch 112/150
48/51  0s 2ms/step - accuracy: 0.8744 - loss: 0.3588
Epoch 112: val_loss improved from 0.33734 to 0.31969, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 4ms/step - accuracy: 0.8733 - loss: 0.3600 - val_accuracy: 0.8953 - val_loss: 0.3197
Epoch 113/150
46/51  0s 1ms/step - accuracy: 0.8585 - loss: 0.3823
Epoch 113: val_loss did not improve from 0.31969
51/51  0s 2ms/step - accuracy: 0.8591 - loss: 0.3814 - val_accuracy: 0.8903 - val_loss: 0.3572
Epoch 114/150
43/51  0s 1ms/step - accuracy: 0.8893 - loss: 0.3259
Epoch 114: val_loss did not improve from 0.31969
51/51  0s 2ms/step - accuracy: 0.8867 - loss: 0.3300 - val_accuracy: 0.9002 - val_loss: 0.3432
Epoch 115/150
37/51  0s 1ms/step - accuracy: 0.8680 - loss: 0.3237
Epoch 115: val_loss did not improve from 0.31969
51/51  0s 2ms/step - accuracy: 0.8683 - loss: 0.3280 - val_accuracy: 0.8653 - val_loss: 0.4178
Epoch 116/150
36/51  0s 1ms/step - accuracy: 0.8447 - loss: 0.3964
Epoch 116: val_loss did not improve from 0.31969
51/51  0s 2ms/step - accuracy: 0.8490 - loss: 0.3885 - val_accuracy: 0.9052 - val_loss: 0.3244
Epoch 117/150
46/51  0s 1ms/step - accuracy: 0.8534 - loss: 0.3843
Epoch 117: val_loss did not improve from 0.31969
51/51  0s 3ms/step - accuracy: 0.8540 - loss: 0.3847 - val_accuracy: 0.8953 - val_loss: 0.3671
Epoch 118/150
41/51  0s 1ms/step - accuracy: 0.8619 - loss: 0.3878
Epoch 118: val_loss did not improve from 0.31969
51/51  0s 2ms/step - accuracy: 0.8610 - loss: 0.3871 - val_accuracy: 0.8903 - val_loss: 0.3389
Epoch 119/150
36/51  0s 1ms/step - accuracy: 0.8539 - loss: 0.3830
Epoch 119: val_loss did not improve from 0.31969
51/51  0s 2ms/step - accuracy: 0.8590 - loss: 0.3736 - val_accuracy: 0.9077 - val_loss: 0.3336

Epoch 120/150
38/51  0s 1ms/step - accuracy: 0.8614 - loss: 0.3886
Epoch 120: val_loss did not improve from 0.31969
51/51  0s 3ms/step - accuracy: 0.8606 - loss: 0.3870 - val_accuracy: 0.8953 - val_loss: 0.3613
Epoch 121/150
43/51  0s 1ms/step - accuracy: 0.8675 - loss: 0.3824
Epoch 121: val_loss did not improve from 0.31969
51/51  0s 2ms/step - accuracy: 0.8661 - loss: 0.3817 - val_accuracy: 0.8628 - val_loss: 0.3696
Epoch 122/150
31/51  0s 2ms/step - accuracy: 0.8377 - loss: 0.4009
Epoch 122: val_loss did not improve from 0.31969
51/51  0s 2ms/step - accuracy: 0.8451 - loss: 0.3918 - val_accuracy: 0.8878 - val_loss: 0.3288
Epoch 123/150
43/51  0s 1ms/step - accuracy: 0.8449 - loss: 0.4090
Epoch 123: val_loss did not improve from 0.31969
51/51  0s 3ms/step - accuracy: 0.8451 - loss: 0.4089 - val_accuracy: 0.8778 - val_loss: 0.3235
Epoch 124/150
43/51  0s 1ms/step - accuracy: 0.8741 - loss: 0.3822
Epoch 124: val_loss did not improve from 0.31969
51/51  0s 2ms/step - accuracy: 0.8730 - loss: 0.3808 - val_accuracy: 0.8878 - val_loss: 0.3222
Epoch 125/150
43/51  0s 1ms/step - accuracy: 0.8724 - loss: 0.3345
Epoch 125: val_loss did not improve from 0.31969
51/51  0s 3ms/step - accuracy: 0.8717 - loss: 0.3362 - val_accuracy: 0.8978 - val_loss: 0.3446
Epoch 126/150
47/51  0s 1ms/step - accuracy: 0.8554 - loss: 0.3589
Epoch 126: val_loss did not improve from 0.31969
51/51  0s 2ms/step - accuracy: 0.8548 - loss: 0.3615 - val_accuracy: 0.8504 - val_loss: 0.4054
Epoch 127/150
39/51  0s 1ms/step - accuracy: 0.8429 - loss: 0.4493
Epoch 127: val_loss improved from 0.31969 to 0.31810, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.8448 - loss: 0.4370 - val_accuracy: 0.8953 - val_loss: 0.3181
Epoch 128/150
24/51  0s 2ms/step - accuracy: 0.8403 - loss: 0.4237
Epoch 128: val_loss did not improve from 0.31810
51/51  0s 2ms/step - accuracy: 0.8508 - loss: 0.4072 - val_accuracy: 0.8928 - val_loss: 0.3236
Epoch 129/150
46/51  0s 1ms/step - accuracy: 0.8811 - loss: 0.3632
Epoch 129: val_loss did not improve from 0.31810
51/51  0s 2ms/step - accuracy: 0.8804 - loss: 0.3623 - val_accuracy: 0.9027 - val_loss: 0.3437
Epoch 130/150
39/51  0s 1ms/step - accuracy: 0.8932 - loss: 0.3156
Epoch 130: val_loss did not improve from 0.31810
51/51  0s 2ms/step - accuracy: 0.8913 - loss: 0.3184 - val_accuracy: 0.8803 - val_loss: 0.3514
Epoch 131/150
32/51  0s 2ms/step - accuracy: 0.8471 - loss: 0.4151
Epoch 131: val_loss did not improve from 0.31810
51/51  0s 4ms/step - accuracy: 0.8547 - loss: 0.4015 - val_accuracy: 0.8853 - val_loss: 0.3427
Epoch 132/150
31/51  0s 2ms/step - accuracy: 0.8792 - loss: 0.3355
Epoch 132: val_loss improved from 0.31810 to 0.31356, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 3ms/step - accuracy: 0.8775 - loss: 0.3399 - val_accuracy: 0.8953 - val_loss: 0.3136
Epoch 133/150
43/51  0s 1ms/step - accuracy: 0.8780 - loss: 0.3313
Epoch 133: val_loss did not improve from 0.31356
51/51  0s 2ms/step - accuracy: 0.8796 - loss: 0.3279 - val_accuracy: 0.8828 - val_loss: 0.3236
Epoch 134/150
44/51  0s 1ms/step - accuracy: 0.8802 - loss: 0.3090
Epoch 134: val_loss did not improve from 0.31356
51/51  0s 3ms/step - accuracy: 0.8794 - loss: 0.3127 - val_accuracy: 0.8928 - val_loss: 0.3414
Epoch 135/150
35/51  0s 2ms/step - accuracy: 0.8676 - loss: 0.3291
Epoch 135: val_loss did not improve from 0.31356
51/51  0s 2ms/step - accuracy: 0.8694 - loss: 0.3299 - val_accuracy: 0.8953 - val_loss: 0.3185
Epoch 136/150


40/51  0s 1ms/step - accuracy: 0.8790 - loss: 0.3142
Epoch 136: val_loss did not improve from 0.31356
51/51  0s 2ms/step - accuracy: 0.8776 - loss: 0.3176 - val_accuracy: 0.8853 - val_loss: 0.3779
Epoch 137/150
40/51  0s 1ms/step - accuracy: 0.8837 - loss: 0.3114
Epoch 137: val_loss did not improve from 0.31356
51/51  0s 2ms/step - accuracy: 0.8843 - loss: 0.3105 - val_accuracy: 0.8903 - val_loss: 0.3443
Epoch 138/150
27/51  0s 3ms/step - accuracy: 0.8869 - loss: 0.3012
Epoch 138: val_loss did not improve from 0.31356
51/51  0s 3ms/step - accuracy: 0.8787 - loss: 0.3179 - val_accuracy: 0.8928 - val_loss: 0.3752
Epoch 139/150
43/51  0s 1ms/step - accuracy: 0.8505 - loss: 0.3914
Epoch 139: val_loss did not improve from 0.31356
51/51  0s 2ms/step - accuracy: 0.8535 - loss: 0.3846 - val_accuracy: 0.8878 - val_loss: 0.3505
Epoch 140/150
43/51  0s 1ms/step - accuracy: 0.8705 - loss: 0.3484
Epoch 140: val_loss did not improve from 0.31356
51/51  0s 2ms/step - accuracy: 0.8706 - loss: 0.3462 - val_accuracy: 0.8928 - val_loss: 0.3250
Epoch 141/150
39/51  0s 1ms/step - accuracy: 0.8746 - loss: 0.3370
Epoch 141: val_loss did not improve from 0.31356
51/51  0s 2ms/step - accuracy: 0.8749 - loss: 0.3355 - val_accuracy: 0.9002 - val_loss: 0.3402
Epoch 142/150
42/51  0s 1ms/step - accuracy: 0.8777 - loss: 0.3379
Epoch 142: val_loss did not improve from 0.31356
51/51  0s 2ms/step - accuracy: 0.8776 - loss: 0.3406 - val_accuracy: 0.8978 - val_loss: 0.3233
Epoch 143/150
33/51  0s 2ms/step - accuracy: 0.8529 - loss: 0.3707
Epoch 143: val_loss did not improve from 0.31356
51/51  0s 3ms/step - accuracy: 0.8581 - loss: 0.3632 - val_accuracy: 0.8579 - val_loss: 0.3960
Epoch 144/150
46/51  0s 1ms/step - accuracy: 0.8429 - loss: 0.4176
Epoch 144: val_loss did not improve from 0.31356
51/51  0s 2ms/step - accuracy: 0.8454 - loss: 0.4117 - val_accuracy: 0.8903 - val_loss: 0.3272
Epoch 145/150
45/51  0s 1ms/step - accuracy: 0.8815 - loss: 0.3589
Epoch 145: val_loss improved from 0.31356 to 0.30762, saving model to saved_models/Asthma_audioclassification.keras
51/51  0s 4ms/step - accuracy: 0.8817 - loss: 0.3566 - val_accuracy: 0.9052 - val_loss: 0.3076
Epoch 146/150
39/51  0s 1ms/step - accuracy: 0.8735 - loss: 0.3328
Epoch 146: val_loss did not improve from 0.30762
51/51  0s 2ms/step - accuracy: 0.8727 - loss: 0.3307 - val_accuracy: 0.8978 - val_loss: 0.3382
Epoch 147/150
37/51  0s 1ms/step - accuracy: 0.8843 - loss: 0.2964
Epoch 147: val_loss did not improve from 0.30762
51/51  0s 2ms/step - accuracy: 0.8850 - loss: 0.2965 - val_accuracy: 0.8978 - val_loss: 0.3361
Epoch 148/150
25/51  0s 2ms/step - accuracy: 0.8851 - loss: 0.3296
Epoch 148: val_loss did not improve from 0.30762
51/51  0s 2ms/step - accuracy: 0.8714 - loss: 0.3589 - val_accuracy: 0.8878 - val_loss: 0.3469
Epoch 149/150
36/51  0s 1ms/step - accuracy: 0.8650 - loss: 0.3601
Epoch 149: val_loss did not improve from 0.30762
51/51  0s 2ms/step - accuracy: 0.8653 - loss: 0.3586 - val_accuracy: 0.9077 - val_loss: 0.3644
Epoch 150/150
43/51  0s 1ms/step - accuracy: 0.8573 - loss: 0.3610
Epoch 150: val_loss did not improve from 0.30762
51/51  0s 2ms/step - accuracy: 0.8597 - loss: 0.3550 - val_accuracy: 0.9002 - val_loss: 0.3848
Training time 0:00:27.224420

```
In [58]: model.evaluate(X_test,y_test,verbose=0)
```

```
Out[58]: [0.3847678303718567, 0.9002493619918823]
```


```
In [59]: filename="Asthma Detection Dataset Version 2\\asthma\\Asthma_RS_20 (1).wav"
y, sr = librosa.load(filename, duration=3, offset=0.5)
```

```
mfcc = np.mean(librosa.feature.mfcc(y=y, sr=sr, n_mfcc=40).T, axis=0)
mfcc=mfcc.reshape(1,-1)
print(mfcc)
print(mfcc.shape)
predicted_label=np.argmax(model.predict(mfcc), axis=-1)
print(predicted_label)
prediction_class = labelencoder.inverse_transform(predicted_label)
prediction_class
```

```
[[-548.83496    50.119514   45.78449    43.384907   40.99531
  37.92376    34.31658    30.903397   27.728077   24.546072
  21.37009    18.457767   15.855426   13.384048   11.001728
   8.931421    7.295938    5.979423    4.9392314   4.3285704
   4.1767793   4.285747    4.5153503    4.891299    5.3693304
   5.767992    5.9868765    6.0878544    6.109491    5.9857993
   5.6918344    5.316192    4.9219117    4.4640946    3.9046216
   3.3001502    2.7046251    2.0866601    1.4203374    0.7722891]]
(1, 40)
1/1  0s 60ms/step
[0]
```


Out[59]: array(['asthma'], dtype='<U9')

```
In [60]: filename="Asthma Detection Dataset Version 2\Bronchial\Bronchial_Sc_1 (1).wav"
y, sr = librosa.load(filename, duration=3, offset=0.5)
mfcc = np.mean(librosa.feature.mfcc(y=y, sr=sr, n_mfcc=40).T, axis=0)
mfcc=mfcc.reshape(1,-1)
print(mfcc)
print(mfcc.shape)
predicted_label=np.argmax(model.predict(mfcc), axis=-1)
print(predicted_label)
prediction_class = labelencoder.inverse_transform(predicted_label)
prediction_class
```

```
[[-3.8983014e+02  1.8178264e+02  2.1946053e+01 -1.5424136e+01
  1.3209252e+01  1.8196669e+01  7.7502036e+00  6.2696522e-01
  6.6663593e-01  5.2164898e+00  9.5117359e+00  6.4938064e+00
 -2.1969423e+00  4.3192797e+00  4.2819500e+00  3.9133751e+00
  1.8397973e-01  9.0698242e+00  3.1407807e+00  4.5960984e+00
 -2.8902512e+00  2.9254174e+00 -1.1361605e-01  6.3802347e+00
 -9.9346149e-01  5.8656082e+00  3.1182034e+00  6.5837083e+00
 -2.4030316e+00  1.1249828e+00 -2.7190251e+00  3.4650650e+00
  4.3189314e-01  2.8748198e+00 -2.7851725e+00  1.3261629e+00
  2.7799010e+00  3.6490676e+00 -2.2812161e+00 -4.8443222e-01]]
(1, 40)
1/1  0s 22ms/step
[1]
```

Out[60]: array(['bronchial'], dtype='<U9')

```
In [61]: filename="Asthma Detection Dataset Version 2\Bronchial\Bronchial_Sc_1 (8).wav"
y, sr = librosa.load(filename, duration=3, offset=0.5)
mfcc = np.mean(librosa.feature.mfcc(y=y, sr=sr, n_mfcc=40).T, axis=0)
mfcc=mfcc.reshape(1,-1)
print(mfcc)
print(mfcc.shape)
predicted_label=np.argmax(model.predict(mfcc), axis=-1)
print(predicted_label)
prediction_class = labelencoder.inverse_transform(predicted_label)
prediction_class
```

```
[[-4.5608768e+02  1.7521689e+02  5.2230350e+01 -1.3782601e+00
 -4.3475981e+00  3.0727172e+00 -6.5864635e-01  2.1587160e+00
  1.4168003e+01  2.2654135e+01  1.4551707e+01  3.4448764e+00
  4.7594452e-01  8.8251476e+00  8.9229660e+00  5.2574186e+00
 -1.7476784e+00  2.4893077e-01  2.7135258e+00  7.2230697e+00
  5.8852682e+00  7.4056950e+00  2.2178140e+00  1.3237664e+00
  9.3586162e-02  2.8307562e+00  8.4737867e-01  1.9972440e+00
  1.8681623e+00  5.0061994e+00  1.6391280e+00  3.8139215e-01
 -1.1640850e+00  1.9350014e+00  1.4548314e+00  1.4240146e+00
 -7.2621983e-01  9.7914219e-01  1.7796893e+00  3.4094405e+00]]
(1, 40)
1/1  0s 38ms/step
[1]
```

Out[61]: array(['bronchial'], dtype='<U9')

```
In [62]: filename="Asthma Detection Dataset Version 2\pneumonia\Pneumonia_10T (5).wav"
y, sr = librosa.load(filename, duration=3, offset=0.5)
mfcc = np.mean(librosa.feature.mfcc(y=y, sr=sr, n_mfcc=40).T, axis=0)
mfcc=mfcc.reshape(1,-1)
print(mfcc)
print(mfcc.shape)
predicted_label=np.argmax(model.predict(mfcc), axis=-1)
print(predicted_label)
prediction_class = labelencoder.inverse_transform(predicted_label)
```



```
prediction_class
```

```
[[-385.1813    113.2246    87.42339    64.046234    45.475803  
  31.254955    19.582722    10.06814    3.4867895    1.1309445  
   2.2522495    4.13407     5.300241    5.9561915    6.156539  
   5.829645    5.2794957    4.7358866    4.2662005    3.8962717  
   3.2000952    2.0528636    1.4345835    2.102308    3.0735042  
   3.2031195    2.614724    1.9900856    1.740515    1.8590331  
   1.8391153    1.2675236    0.5058664    0.3985395    0.7895945  
   0.881756    0.6754371    1.0320972    2.1741264    3.1455822]]
```

```
(1, 40)
```

```
1/1  0s 21ms/step
```

```
[4]
```

```
Out[62]: array(['pneumonia'], dtype='<U9')
```

```
In [ ]: filename=  
y, sr = librosa.load(filename, duration=3, offset=0.5)  
mfcc = np.mean(librosa.feature.mfcc(y=y, sr=sr, n_mfcc=40).T, axis=0)  
mfcc=mfcc.reshape(1,-1)  
print(mfcc)  
print(mfcc.shape)  
predicted_label=np.argmax(model.predict(mfcc), axis=-1)  
print(predicted_label)  
prediction_class = labelencoder.inverse_transform(predicted_label)  
prediction_class
```

```
Loading [MathJax]/extensions/Safe.js
```