SPRINT 2: Classification of Arrhythmia by Using Deep Learning With2-D ECG Spectral Image Representation

Team ID: PNT2022TMID27741

Team Members: Shakthi, Shruthi, Chelsia stella, Lavanya

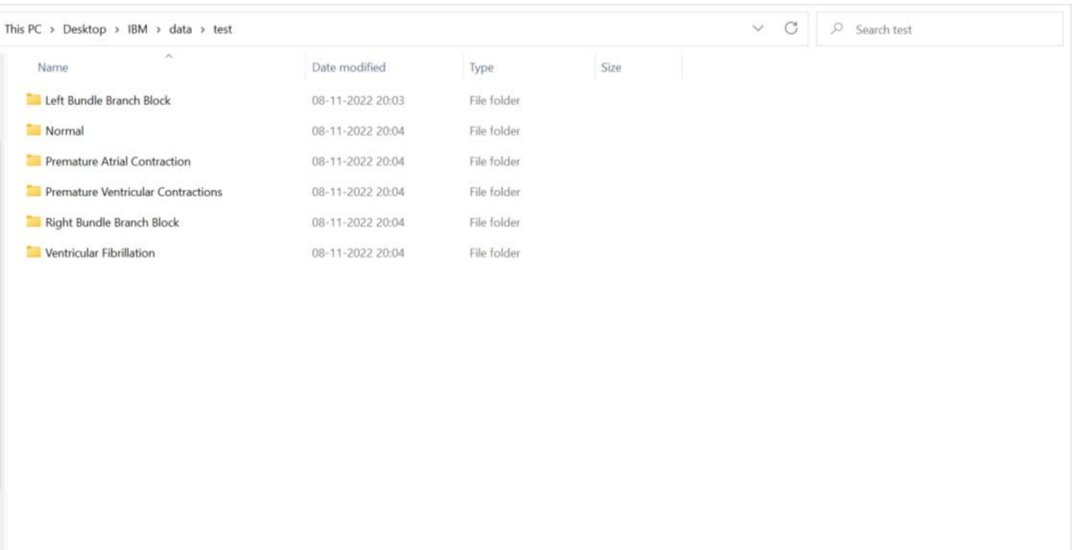
Code: Updated in GitHub in the Deliverables section in Sprint 2 folder.

Description of USN and Screenshots:

USN-4:

As a user, I want quality data to be collected for the purposes of training the model. Also, image processing methods must be employed to pre-process the dataset.

Screenshot:



Name	Date modified	Type	Size
Left Bundle Branch Block	08-11-2022 20:03	File folder	
Normal	08-11-2022 20:04	File folder	
Premature Atrial Contraction	08-11-2022 20:04	File folder	
Premature Ventricular Contractions	08-11-2022 20:04	File folder	
Right Bundle Branch Block	08-11-2022 20:04	File folder	
Ventricular Fibrillation	08-11-2022 20:04	File folder	

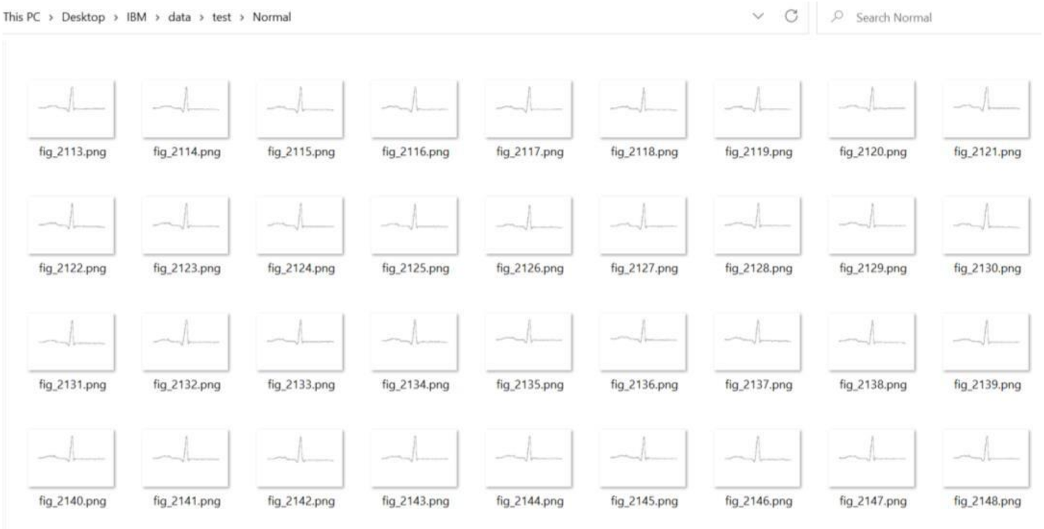


Image Split:

Left Bundle Branch Block – 504 images

Normal – 7436 images

Premature Atrial Contraction – 2054 images

Premature Ventricular Contractions – 2759 images

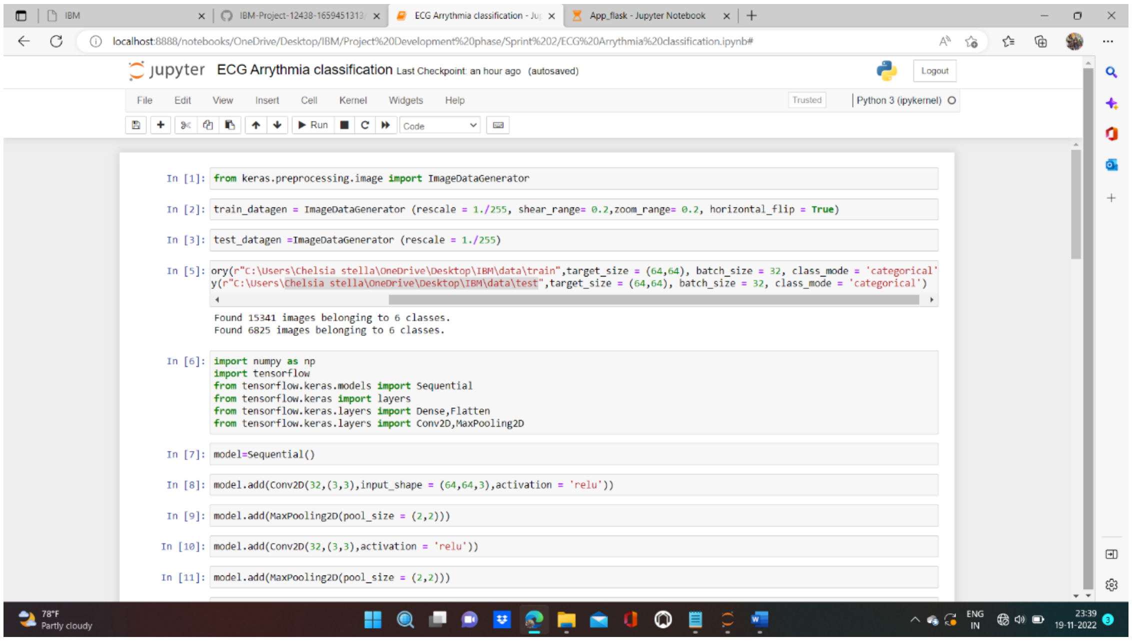
Right Bundle Branch Block – 2239 images

Ventricular Fibrillation – 439 images

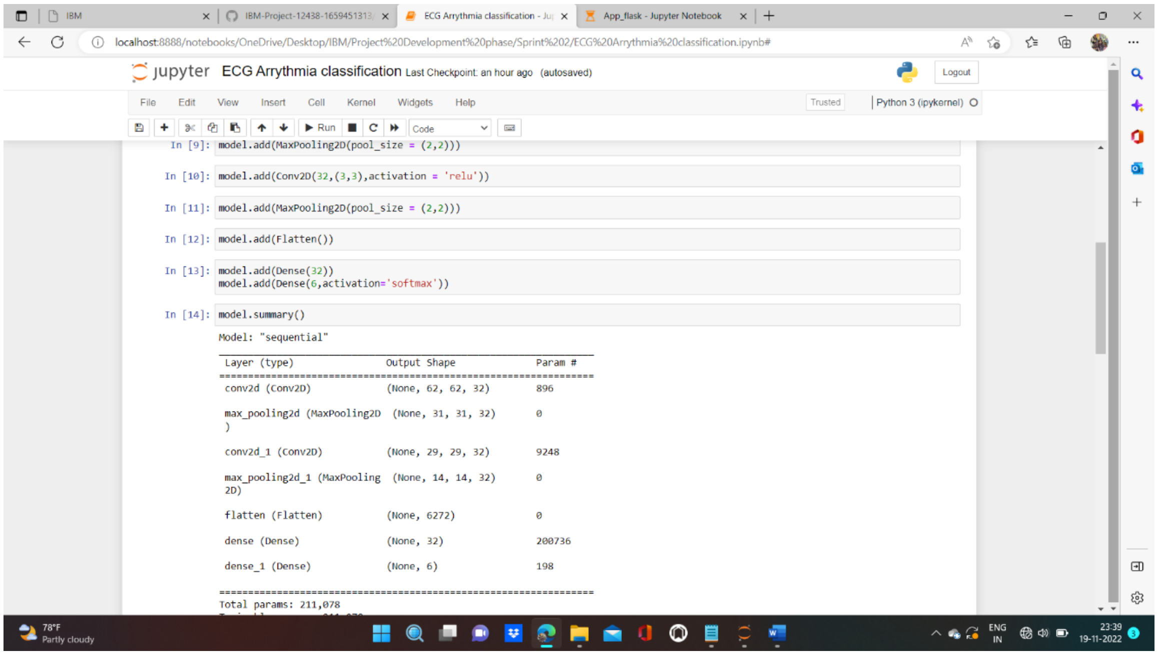
For reducing skewness in the dataset, ImageDataGenerator class was used for both processing and handling with data imbalance

As a user, I want the ML model to be as accurate as possible.

Screenshot:


IBM
IBM-Project-12438-1659451313
ECG Arrythmia classification - Ju
App_flask - Jupyter Notebook
localhost:8888/notebooks/OneDrive/Desktop/IBM/Project%20Development%20phase/Sprint%202/ECG%20Arrythmia%20classification.ipynb#
jupyter
ECG Arrythmia classification Last Checkpoint: an hour ago (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Trusted
Python 3 (ipykernel)
Run
Code
78°F
Partly cloudy
ENG
IN
23:39
19-11-2022

```
In [1]: from keras.preprocessing.image import ImageDataGenerator

In [2]: train_datagen = ImageDataGenerator (rescale = 1./255, shear_range= 0.2,zoom_range= 0.2, horizontal_flip = True)

In [3]: test_datagen =ImageDataGenerator (rescale = 1./255)

In [5]: ory(r"C:\Users\Chelsia stella\OneDrive\Desktop\IBM\data\train",target_size = (64,64), batch_size = 32, class_mode = 'categorical'
        y(r"C:\Users\Chelsia stella\OneDrive\Desktop\IBM\data\test",target_size = (64,64), batch_size = 32, class_mode = 'categorical')

        Found 15341 images belonging to 6 classes.
        Found 6825 images belonging to 6 classes.

In [6]: import numpy as np
        import tensorflow
        from tensorflow.keras.models import Sequential
        from tensorflow.keras import layers
        from tensorflow.keras.layers import Dense,Flatten
        from tensorflow.keras.layers import Conv2D,MaxPooling2D

In [7]: model=Sequential()

In [8]: model.add(Conv2D(32,(3,3),input_shape = (64,64,3),activation = 'relu'))

In [9]: model.add(MaxPooling2D(pool_size = (2,2)))

In [10]: model.add(Conv2D(32,(3,3),activation = 'relu'))

In [11]: model.add(MaxPooling2D(pool_size = (2,2)))
```

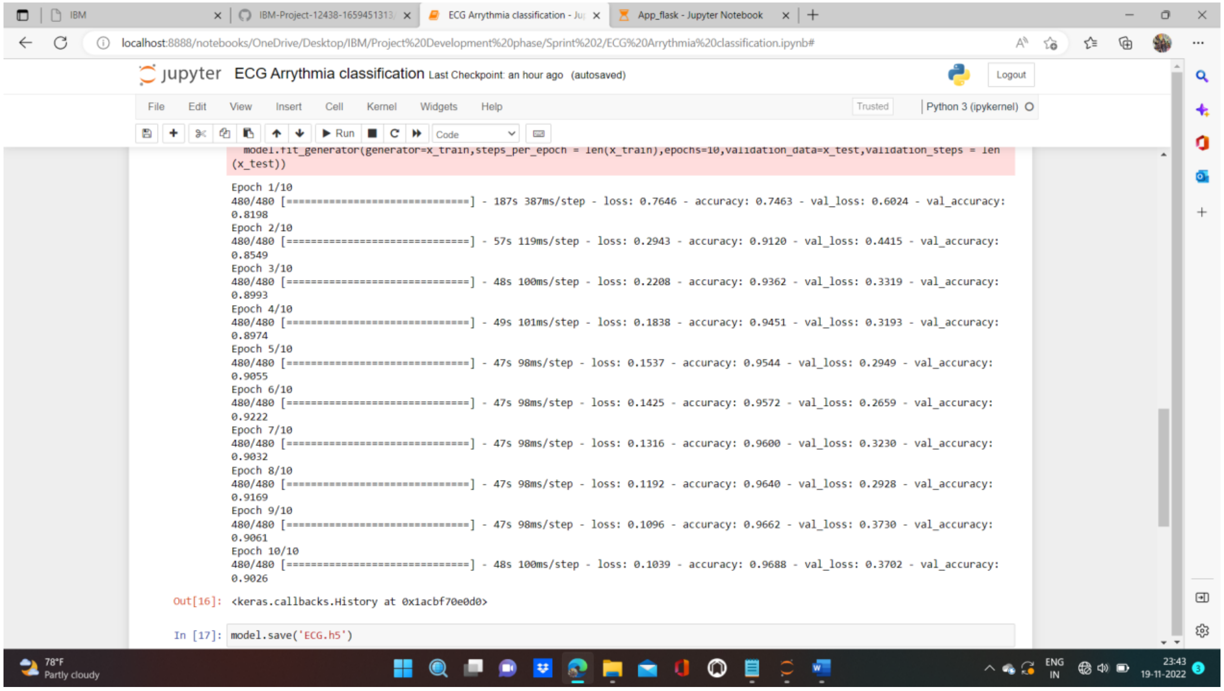

IBM
IBM-Project-12438-1659451313
ECG Arrythmia classification - Ju
App_flask - Jupyter Notebook
localhost:8888/notebooks/OneDrive/Desktop/IBM/Project%20Development%20phase/Sprint%202/ECG%20Arrythmia%20classification.ipynb#
jupyter
ECG Arrythmia classification Last Checkpoint: an hour ago (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Trusted
Python 3 (ipykernel)
Run
Code
78°F
Partly cloudy
ENG
IN
23:39
19-11-2022

```
In [9]: model.add(MaxPooling2D(pool_size = (2,2)))

In [10]: model.add(Conv2D(32,(3,3),activation = 'relu'))

In [11]: model.add(MaxPooling2D(pool_size = (2,2)))
```

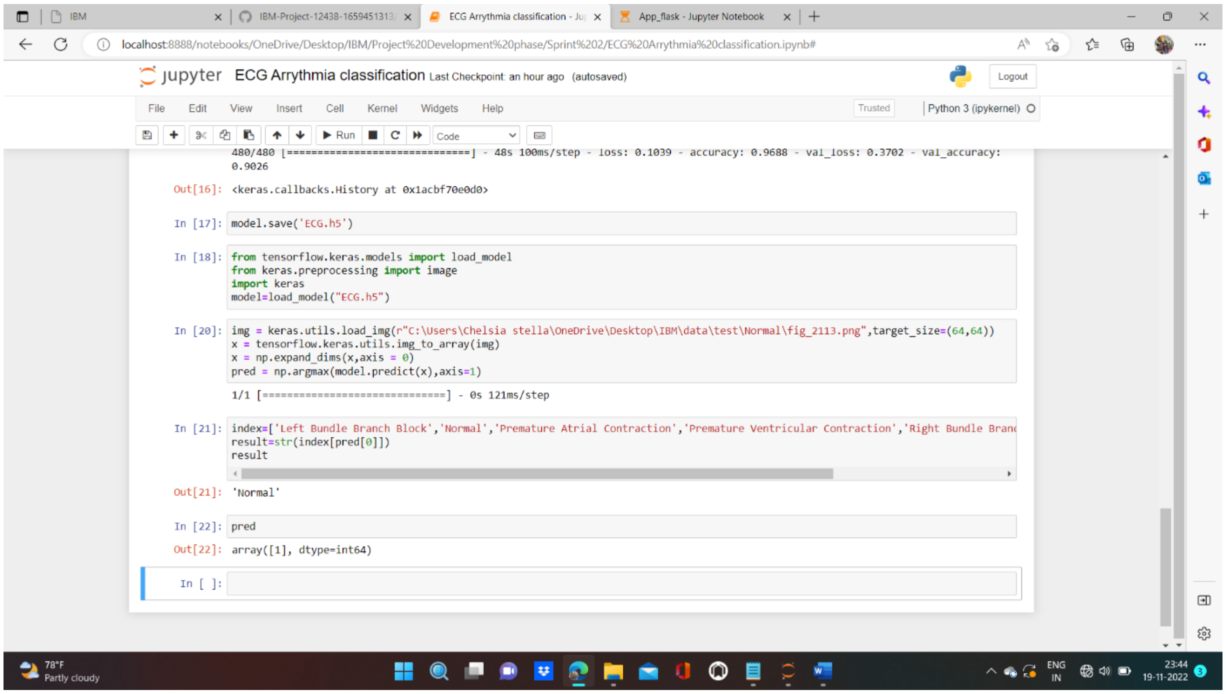
```
In [12]: model.add(Flatten())

In [13]: model.add(Dense(32))
         model.add(Dense(6,activation='softmax'))

In [14]: model.summary()

         Model: "sequential"
         _________________________________________________________________
          Layer (type)                Output Shape              Param #
         =================================================================
          conv2d (Conv2D)             (None, 62, 62, 32)        896

          max_pooling2d (MaxPooling2D  (None, 31, 31, 32)       0
          )

          conv2d_1 (Conv2D)           (None, 29, 29, 32)        9248

          max_pooling2d_1 (MaxPooling  (None, 14, 14, 32)       0
          2D)

          flatten (Flatten)           (None, 6272)              0

          dense (Dense)               (None, 32)                200736

          dense_1 (Dense)             (None, 6)                 198

         =================================================================
         Total params: 211,078
```

Model Architecture:

```
Model: "sequential"
_________________________________________________________________

 Layer (type)           Output Shape          Param #
=================================================================
 conv2d (Conv2D)          (None, 62, 62, 32)      896

 max_pooling2d (MaxPooling2D  (None, 31, 31, 32))   0

 conv2d_1 (Conv2D)        (None, 29, 29, 32)      9248

 max_pooling2d_1 (MaxPooling  (None, 14, 14, 32) 2D)   0

 flatten (Flatten)        (None, 6272)          0

 dense (Dense)            (None, 32)            200736
dense_1 (Dense)           (None, 6)             198

=================================================================
Total params: 211,078

Trainable params: 211,078

Non-trainable params: 0
```


IBM
IBM-Project-12438-1659451313
ECG Arrythmia classification - Ju
App_flask - Jupyter Notebook
localhost:8888/notebooks/OneDrive/Desktop/IBM/Project%20Development%20phase/Sprint%202/ECG%20Arrythmia%20classification.ipynb#
jupyter
ECG Arrythmia classification Last Checkpoint: an hour ago (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Trusted
Python 3 (ipykernel)
Run
Code
78°F
Partly cloudy
ENG
IN
23:43
19-11-2022

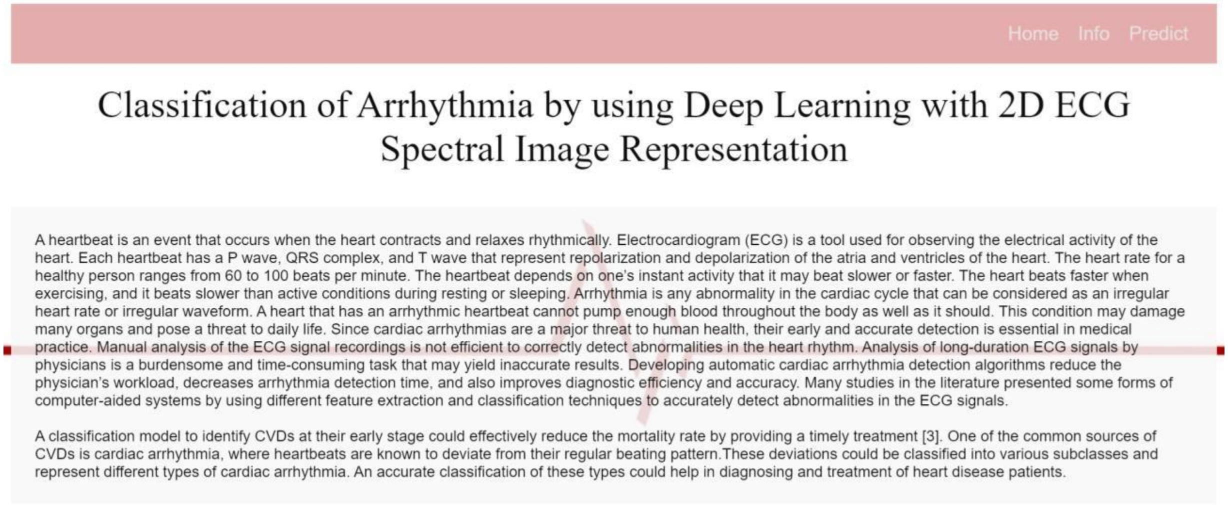
```
  model.fit_generator(generator=x_train,steps_per_epoch = len(x_train),epochs=10,validation_data=x_test,validation_steps = len
(x_test))

Epoch 1/10
480/480 [==============================] - 187s 387ms/step - loss: 0.7646 - accuracy: 0.7463 - val_loss: 0.6024 - val_accuracy:
0.8198
Epoch 2/10
480/480 [==============================] - 57s 119ms/step - loss: 0.2943 - accuracy: 0.9120 - val_loss: 0.4415 - val_accuracy:
0.8549
Epoch 3/10
480/480 [==============================] - 48s 100ms/step - loss: 0.2208 - accuracy: 0.9362 - val_loss: 0.3319 - val_accuracy:
0.8993
Epoch 4/10
480/480 [==============================] - 49s 101ms/step - loss: 0.1838 - accuracy: 0.9451 - val_loss: 0.3193 - val_accuracy:
0.8974
Epoch 5/10
480/480 [==============================] - 47s 98ms/step - loss: 0.1537 - accuracy: 0.9544 - val_loss: 0.2949 - val_accuracy:
0.9055
Epoch 6/10
480/480 [==============================] - 47s 98ms/step - loss: 0.1425 - accuracy: 0.9572 - val_loss: 0.2659 - val_accuracy:
0.9222
Epoch 7/10
480/480 [==============================] - 47s 98ms/step - loss: 0.1316 - accuracy: 0.9600 - val_loss: 0.3230 - val_accuracy:
0.9032
Epoch 8/10
480/480 [==============================] - 47s 98ms/step - loss: 0.1192 - accuracy: 0.9640 - val_loss: 0.2928 - val_accuracy:
0.9169
Epoch 9/10
480/480 [==============================] - 47s 98ms/step - loss: 0.1096 - accuracy: 0.9662 - val_loss: 0.3730 - val_accuracy:
0.9061
Epoch 10/10
480/480 [==============================] - 48s 100ms/step - loss: 0.1039 - accuracy: 0.9688 - val_loss: 0.3702 - val_accuracy:
0.9026

Out[16]: <keras.callbacks.History at 0x1acbf70e0d0>

In [17]: model.save('ECG.h5')
```



IBM
IBM-Project-12438-1659451313
ECG Arrythmia classification - Ju
App_flask - Jupyter Notebook
localhost:8888/notebooks/OneDrive/Desktop/IBM/Project%20Development%20phase/Sprint%202/ECG%20Arrythmia%20classification.ipynb#
jupyter
ECG Arrythmia classification Last Checkpoint: an hour ago (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Trusted
Python 3 (ipykernel)
Run
Code
78°F
Partly cloudy
ENG
IN
23:44
19-11-2022

```
480/480 [==============================] - 48s 100ms/step - loss: 0.1039 - accuracy: 0.9688 - val_loss: 0.3702 - val_accuracy:
0.9026

Out[16]: <keras.callbacks.History at 0x1acbf70e0d0>

In [17]: model.save('ECG.h5')

In [18]: from tensorflow.keras.models import load_model
         from keras.preprocessing import image
         import keras
         model=load_model("ECG.h5")

In [20]: img = keras.utils.load_img(r"C:\Users\Chelsia stella\OneDrive\Desktop\IBM\data\test\Normal\fig_2113.png",target_size=(64,64))
         x = tensorflow.keras.utils.img_to_array(img)
         x = np.expand_dims(x,axis = 0)
         pred = np.argmax(model.predict(x),axis=1)

         1/1 [==============================] - 0s 121ms/step

In [21]: index=['Left Bundle Branch Block','Normal','Premature Atrial Contraction','Premature Ventricular Contraction','Right Bundle Branc
         result=str(index[pred[0]])
         result

Out[21]: 'Normal'

In [22]: pred

Out[22]: array([1], dtype=int64)

In [ ]:
```

As a user, I can view my home page in the dashboard. I will get the full idea of ECG Arrhythmia classification using CNN where the details of the webpage will be given and info about different CVDs are provided. The homepage must properly define the Arrhythmia, its causes and effects and understand how the application helps in solving the problem.



Classification of Arrhythmia by using Deep Learning with 2D ECG Spectral Image Representation

A heartbeat is an event that occurs when the heart contracts and relaxes rhythmically. Electrocardiogram (ECG) is a tool used for observing the electrical activity of the heart. Each heartbeat has a P wave, QRS complex, and T wave that represent repolarization and depolarization of the atria and ventricles of the heart. The heart rate for a healthy person ranges from 60 to 100 beats per minute. The heartbeat depends on one's instant activity that it may beat slower or faster. The heart beats faster when exercising, and it beats slower than active conditions during resting or sleeping. Arrhythmia is any abnormality in the cardiac cycle that can be considered as an irregular heart rate or irregular waveform. A heart that has an arrhythmic heartbeat cannot pump enough blood throughout the body as well as it should. This condition may damage many organs and pose a threat to daily life. Since cardiac arrhythmias are a major threat to human health, their early and accurate detection is essential in medical practice. Manual analysis of the ECG signal recordings is not efficient to correctly detect abnormalities in the heart rhythm. Analysis of long-duration ECG signals by physicians is a burdensome and time-consuming task that may yield inaccurate results. Developing automatic cardiac arrhythmia detection algorithms reduce the physician's workload, decreases arrhythmia detection time, and also improves diagnostic efficiency and accuracy. Many studies in the literature presented some forms of computer-aided systems by using different feature extraction and classification techniques to accurately detect abnormalities in the ECG signals.

A classification model to identify CVDs at their early stage could effectively reduce the mortality rate by providing a timely treatment [3]. One of the common sources of CVDs is cardiac arrhythmia, where heartbeats are known to deviate from their regular beating pattern.These deviations could be classified into various subclasses and represent different types of cardiac arrhythmia. An accurate classification of these types could help in diagnosing and treatment of heart disease patients.