In [11]:

```
!pip install keras
```

```
Requirement already satisfied: keras in c:\users\shadm\anaconda3\lib\site-pa
ckages (2.8.0)
```

In [12]:

```
!pip install tensorflow
```

```
Requirement already satisfied: tensorflow in c:\users\shadm\anaconda3\lib\si
te-packages (2.8.0)
Requirement already satisfied: absl-py>=0.4.0 in c:\users\shadm\anaconda3\li
b\site-packages (from tensorflow) (1.0.0)
Requirement already satisfied: tensorboard<2.9,>=2.8 in c:\users\shadm\anaco
nda3\lib\site-packages (from tensorflow) (2.8.0)
Requirement already satisfied: wrapt>=1.11.0 in c:\users\shadm\anaconda3\lib
\site-packages (from tensorflow) (1.12.1)
Requirement already satisfied: termcolor>=1.1.0 in c:\users\shadm\anaconda3
\lib\site-packages (from tensorflow) (1.1.0)
Requirement already satisfied: opt-einsum>=2.3.2 in c:\users\shadm\anaconda3
\lib\site-packages (from tensorflow) (3.3.0)
Requirement already satisfied: grpcio<2.0,>=1.24.3 in c:\users\shadm\anacond
a3\lib\site-packages (from tensorflow) (1.42.0)
Requirement already satisfied: keras<2.9,>=2.8.0rc0 in c:\users\shadm\anacon
da3\lib\site-packages (from tensorflow) (2.8.0)
Requirement already satisfied: h5py>=2.9.0 in c:\users\shadm\anaconda3\lib\s
ite-packages (from tensorflow) (3.6.0)
Requirement already satisfied: keras-preprocessing>=1.1.1 in c:\users\shadm
\anaconda3\lib\site-packages (from tensorflow) (1.1.2)
Requirement already satisfied: numpy>=1.20 in c:\users\shadm\anaconda3\lib\s
ite-packages (from tensorflow) (1.21.5)
Requirement already satisfied: astunparse>=1.6.0 in c:\users\shadm\anaconda3
\lib\site-packages (from tensorflow) (1.6.3)
Requirement already satisfied: flatbuffers>=1.12 in c:\users\shadm\anaconda3
\lib\site-packages (from tensorflow) (2.0)
Requirement already satisfied: protobuf>=3.9.2 in c:\users\shadm\anaconda3\l
ib\site-packages (from tensorflow) (3.19.1)
Requirement already satisfied: libclang>=9.0.1 in c:\users\shadm\anaconda3\l
ib\site-packages (from tensorflow) (14.0.1)
Requirement already satisfied: setuptools in c:\users\shadm\anaconda3\lib\si
te-packages (from tensorflow) (61.2.0)
Requirement already satisfied: tf-estimator-nightly==2.8.0.dev2021122109 in
c:\users\shadm\anaconda3\lib\site-packages (from tensorflow) (2.8.0.dev20211
22109)
Requirement already satisfied: six>=1.12.0 in c:\users\shadm\anaconda3\lib\s
ite-packages (from tensorflow) (1.16.0)
Requirement already satisfied: tensorflow-io-gcs-filesystem>=0.23.1 in c:\us
ers\shadm\anaconda3\lib\site-packages (from tensorflow) (0.25.0)
Requirement already satisfied: google-pasta>=0.1.1 in c:\users\shadm\anacond
a3\lib\site-packages (from tensorflow) (0.2.0)
Requirement already satisfied: gast>=0.2.1 in c:\users\shadm\anaconda3\lib\s
ite-packages (from tensorflow) (0.5.3)
Requirement already satisfied: typing-extensions>=3.6.6 in c:\users\shadm\an
aconda3\lib\site-packages (from tensorflow) (4.1.1)
Requirement already satisfied: wheel<1.0,>=0.23.0 in c:\users\shadm\anaconda
3\lib\site-packages (from astunparse>=1.6.0->tensorflow) (0.37.1)
Requirement already satisfied: markdown>=2.6.8 in c:\users\shadm\anaconda3\l
ib\site-packages (from tensorboard<2.9,>=2.8->tensorflow) (3.3.4)
Requirement already satisfied: google-auth-oauthlib<0.5,>=0.4.1 in c:\users
\shadm\anaconda3\lib\site-packages (from tensorboard<2.9,>=2.8->tensorflow)
(0.4.6)
Requirement already satisfied: requests<3,>=2.21.0 in c:\users\shadm\anacond
a3\lib\site-packages (from tensorboard<2.9,>=2.8->tensorflow) (2.27.1)
Requirement already satisfied: werkzeug>=0.11.15 in c:\users\shadm\anaconda3
\lib\site-packages (from tensorboard<2.9,>=2.8->tensorflow) (2.0.3)
Requirement already satisfied: tensorboard-data-server<0.7.0,>=0.6.0 in c:\u
```

```
sers\shadm\anaconda3\lib\site-packages (from tensorboard<2.9,>=2.8->tensorfl
ow) (0.6.1)
Requirement already satisfied: google-auth<3,>=1.6.3 in c:\users\shadm\anaco
nda3\lib\site-packages (from tensorboard<2.9,>=2.8->tensorflow) (1.33.0)
Requirement already satisfied: tensorboard-plugin-wit>=1.6.0 in c:\users\sha
dm\anaconda3\lib\site-packages (from tensorboard<2.9,>=2.8->tensorflow) (1.
8.1)
Requirement already satisfied: pyasn1-modules>=0.2.1 in c:\users\shadm\anaco
nda3\lib\site-packages (from google-auth<3,>=1.6.3->tensorboard<2.9,>=2.8->t
ensorflow) (0.2.8)
Requirement already satisfied: cachetools<5.0,>=2.0.0 in c:\users\shadm\anac
onda3\lib\site-packages (from google-auth<3,>=1.6.3->tensorboard<2.9,>=2.8->
tensorflow) (4.2.2)
Requirement already satisfied: rsa<5,>=3.1.4 in c:\users\shadm\anaconda3\lib
\site-packages (from google-auth<3,>=1.6.3->tensorboard<2.9,>=2.8->tensorflo
w) (4.7.2)
Requirement already satisfied: requests-oauthlib>=0.7.0 in c:\users\shadm\an
aconda3\lib\site-packages (from google-auth-oauthlib<0.5,>=0.4.1->tensorboar
d<2.9,>=2.8->tensorflow) (1.3.1)
Requirement already satisfied: pyasn1<0.5.0,>=0.4.6 in c:\users\shadm\anacon
da3\lib\site-packages (from pyasn1-modules>=0.2.1->google-auth<3,>=1.6.3->te
nsorboard<2.9,>=2.8->tensorflow) (0.4.8)
Requirement already satisfied: idna<4,>=2.5 in c:\users\shadm\anaconda3\lib
\site-packages (from requests<3,>=2.21.0->tensorboard<2.9,>=2.8->tensorflow)
(3.3)
Requirement already satisfied: charset-normalizer~=2.0.0 in c:\users\shadm\a
naconda3\lib\site-packages (from requests<3,>=2.21.0->tensorboard<2.9,>=2.8-
>tensorflow) (2.0.4)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\users\shadm\anaco
nda3\lib\site-packages (from requests<3,>=2.21.0->tensorboard<2.9,>=2.8->ten
sorflow) (1.26.9)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\shadm\anaconda
3\lib\site-packages (from requests<3,>=2.21.0->tensorboard<2.9,>=2.8->tensor
flow) (2021.10.8)
Requirement already satisfied: oauthlib>=3.0.0 in c:\users\shadm\anaconda3\l
ib\site-packages (from requests-oauthlib>=0.7.0->google-auth-oauthlib<0.5,>=
0.4.1->tensorboard<2.9,>=2.8->tensorflow) (3.2.0)
```

In [23]:

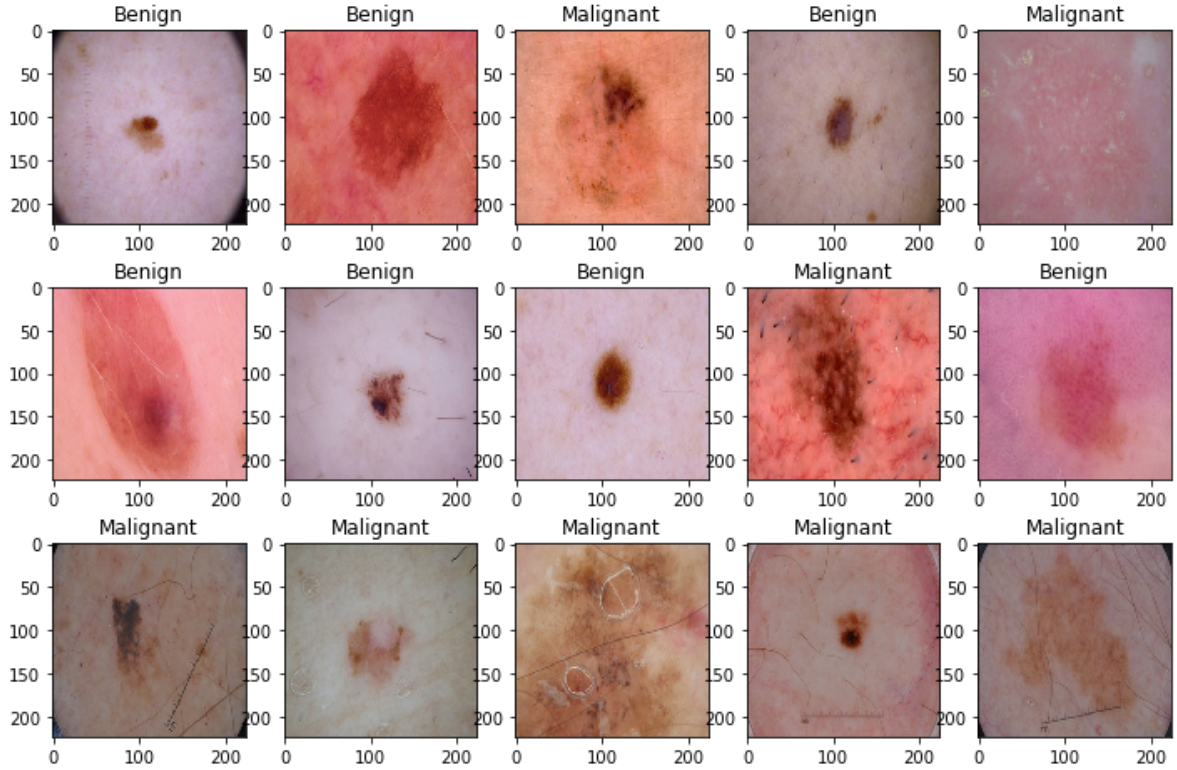
```
import os

%matplotlib inline
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
import os
from glob import glob
import seaborn as sns
from PIL import Image
np.random.seed(11)
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split, KFold, cross_val_score, GridSearchCV
from sklearn.metrics import accuracy_score
import itertools

import keras
from keras.utils.np_utils import to_categorical
from keras.models import Sequential, Model
from keras.layers import Dense, Dropout, Flatten, Conv2D, MaxPool2D
from keras import backend as K
from tensorflow.keras.layers import BatchNormalization
from keras.utils.np_utils import to_categorical
from tensorflow.keras.optimizers import Adam, RMSprop
from keras.preprocessing.image import ImageDataGenerator
from keras.callbacks import ReduceLROnPlateau
from keras.wrappers.scikit_learn import KerasClassifier
from tensorflow.keras.applications.resnet50 import ResNet50
from keras import backend as K
```

In [24]:

```
folder_benign_train = r'C:\Users\shadm\Desktop\Project\Biomedical_Image_Processing_using_De
folder_malignant_train = r'C:\Users\shadm\Desktop\Project\Biomedical_Image_Processing_using

folder_benign_test = r'C:\Users\shadm\Desktop\Project\Biomedical_Image_Processing_using_Dee
folder_malignant_test = r'C:\Users\shadm\Desktop\Project\Biomedical_Image_Processing_using_

read = lambda imname: np.asarray(Image.open(imname).convert("RGB"))

ims_benign = [read(os.path.join(folder_benign_train, filename)) for filename in os.listdir(
X_benign = np.array(ims_benign, dtype='uint8')
ims_malignant = [read(os.path.join(folder_malignant_train, filename)) for filename in os.li
X_malignant = np.array(ims_malignant, dtype='uint8')

ims_benign = [read(os.path.join(folder_benign_test, filename)) for filename in os.listdir(f
X_benign_test = np.array(ims_benign, dtype='uint8')
ims_malignant = [read(os.path.join(folder_malignant_test, filename)) for filename in os.lis
X_malignant_test = np.array(ims_malignant, dtype='uint8')

y_benign = np.zeros(X_benign.shape[0])
y_malignant = np.ones(X_malignant.shape[0])

y_benign_test = np.zeros(X_benign_test.shape[0])
y_malignant_test = np.ones(X_malignant_test.shape[0])

X_train = np.concatenate((X_benign, X_malignant), axis = 0)
y_train = np.concatenate((y_benign, y_malignant), axis = 0)

X_test = np.concatenate((X_benign_test, X_malignant_test), axis = 0)
y_test = np.concatenate((y_benign_test, y_malignant_test), axis = 0)

s = np.arange(X_train.shape[0])
np.random.shuffle(s)
X_train = X_train[s]
y_train = y_train[s]

s = np.arange(X_test.shape[0])
np.random.shuffle(s)
X_test = X_test[s]
y_test = y_test[s]
```

In [25]:

```
w=40
h=30
fig=plt.figure(figsize=(12, 8))
columns = 5
rows = 3

for i in range(1, columns*rows +1):
    ax = fig.add_subplot(rows, columns, i)
    if y_train[i] == 0:
        ax.title.set_text('Benign')
    else:
        ax.title.set_text('Malignant')
    plt.imshow(X_train[i], interpolation='nearest')
plt.show()
```



In [26]:

```
y_train = to_categorical(y_train, num_classes= 2)
y_test = to_categorical(y_test, num_classes= 2)
```

In [27]:

```
X_train = X_train/255.
X_test = X_test/255.
```

In [28]:

```
def build(input_shape= (224,224,3), lr = 1e-3, num_classes= 2,
          init= 'normal', activ= 'relu', optim= 'adam'):
    model = Sequential()
    model.add(Conv2D(64, kernel_size=(3, 3),padding = 'Same',input_shape=input_shape,
                     activation= activ, kernel_initializer='glorot_uniform'))
    model.add(MaxPool2D(pool_size = (2, 2)))
    model.add(Dropout(0.25))

    model.add(Conv2D(64, kernel_size=(3, 3),padding = 'Same',
                     activation =activ, kernel_initializer = 'glorot_uniform'))
    model.add(MaxPool2D(pool_size = (2, 2)))
    model.add(Dropout(0.25))

    model.add(Flatten())
    model.add(Dense(128, activation='relu', kernel_initializer=init))
    model.add(Dense(num_classes, activation='softmax'))
    model.summary()

    if optim == 'rmsprop':
        optimizer = RMSprop(lr=lr)

    else:
        optimizer = Adam(lr=lr)

    model.compile(optimizer = optimizer ,loss = "binary_crossentropy", metrics=["accuracy"]
    return model

learning_rate_reduction = ReduceLROnPlateau(monitor='val_accuracy',
                                            patience=5,
                                            verbose=1,
                                            factor=0.5,
                                            min_lr=1e-7)
```

In [29]:

```
input_shape = (224,224,3)
learning_rate = 1e-5
init = 'normal'
activ = 'relu'
optim = 'adam'
epochs = 5
batch_size = 64

model = build(lr=lr, init= init, activ= activ, optim=optim, input_shape= input_shape)
```

```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d_2 (Conv2D)           (None, 224, 224, 64)      1792

 max_pooling2d_2 (MaxPooling  (None, 112, 112, 64)     0
 2D)

 dropout_2 (Dropout)         (None, 112, 112, 64)      0

 conv2d_3 (Conv2D)           (None, 112, 112, 64)      36928

 max_pooling2d_3 (MaxPooling  (None, 56, 56, 64)       0
 2D)

 dropout_3 (Dropout)         (None, 56, 56, 64)        0

 flatten_1 (Flatten)         (None, 200704)            0

 dense_2 (Dense)             (None, 128)               25690240

 dense_3 (Dense)             (None, 2)                 258

=================================================================
Total params: 25,729,218
Trainable params: 25,729,218
Non-trainable params: 0
_________________________________________________________________
```

In [30]:

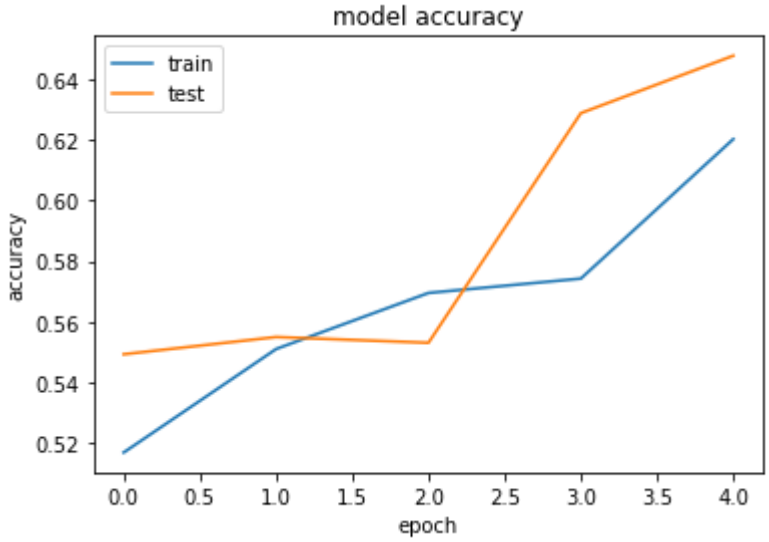
```
history = model.fit(X_train, y_train, validation_split=0.2,
                    epochs= epochs, batch_size= batch_size, verbose=1,
                    callbacks=[learning_rate_reduction]
                   )
```

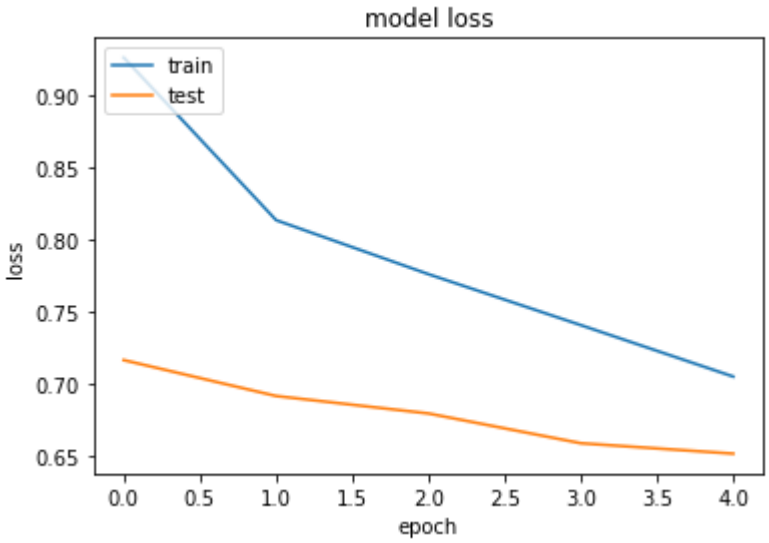
```
Epoch 1/5
33/33 [==============================] - 230s 7s/step - loss: 0.9262 - accur
acy: 0.5168 - val_loss: 0.7165 - val_accuracy: 0.5492 - lr: 1.0000e-05
Epoch 2/5
33/33 [==============================] - 214s 6s/step - loss: 0.8135 - accur
acy: 0.5510 - val_loss: 0.6916 - val_accuracy: 0.5549 - lr: 1.0000e-05
Epoch 3/5
33/33 [==============================] - 205s 6s/step - loss: 0.7761 - accur
acy: 0.5695 - val_loss: 0.6795 - val_accuracy: 0.5530 - lr: 1.0000e-05
Epoch 4/5
33/33 [==============================] - 134s 4s/step - loss: 0.7407 - accur
acy: 0.5742 - val_loss: 0.6588 - val_accuracy: 0.6288 - lr: 1.0000e-05
Epoch 5/5
33/33 [==============================] - 133s 4s/step - loss: 0.7051 - accur
acy: 0.6202 - val_loss: 0.6516 - val_accuracy: 0.6477 - lr: 1.0000e-05
```

In [31]:

```
print(history.history.keys())
plt.plot(history.history['accuracy'])
plt.plot(history.history['val_accuracy'])
plt.title('model accuracy')
plt.ylabel('accuracy')
plt.xlabel('epoch')
plt.legend(['train', 'test'], loc='upper left')
plt.show()
plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])
plt.title('model loss')
plt.ylabel('loss')
plt.xlabel('epoch')
plt.legend(['train', 'test'], loc='upper left')
plt.show()
```

```
dict_keys(['loss', 'accuracy', 'val_loss', 'val_accuracy', 'lr'])
```





In [ ]: