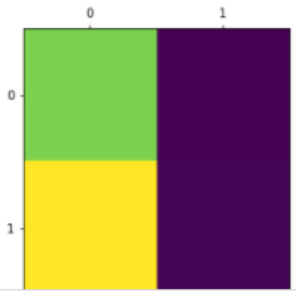
CLASSIFICATION REPORT

1. Binary Classification

❖ No Findings

```
[34] print('test accuracy = ',multi_disease_model.evaluate(test_X,test_y2, verbose=0)[1])

     test accuracy =  0.44970703125

     from sklearn.metrics import roc_auc_score
     print('roc score = ',roc_auc_score(test_y2.astype(int), y_pred))

     roc score =  0.4999277433675637

[39] from sklearn.metrics import classification_report, confusion_matrix
     plt.matshow(confusion_matrix(test_y2, y_pred>0.05839929))
     print(classification_report(test_y2, y_pred>0.05839929, target_names = ['No Findings', 'Findings']))

                   precision    recall  f1-score   support

       No Findings       0.45      0.93      0.61       921
          Findings       0.55      0.07      0.13      1127

          accuracy                           0.46      2048
         macro avg       0.50      0.50      0.37      2048
      weighted avg       0.50      0.46      0.34      2048
```



✓ 0s completed at 5:52 PM

❖ Atelectasis

```
[59] print('test accuracy = ',multi_disease_model.evaluate(test_X,test_y2, verbose=0)[1])
```

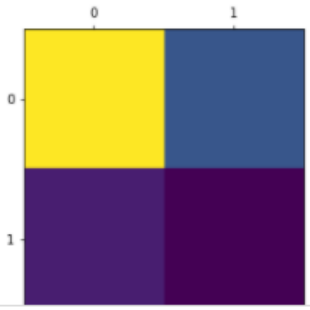
```
test accuracy =  0.8916015625
```

```
from sklearn.metrics import roc_auc_score
print('roc score = ',roc_auc_score(test_y2.astype(int), y_pred))
```

```
roc score =  0.5082603139832056
```

```
[62] from sklearn.metrics import classification_report, confusion_matrix
plt.matshow(confusion_matrix(test_y2, y_pred>0.3849476))
print(classification_report(test_y2, y_pred>0.3849476, target_names = ['Atelectasis', 'No Atelectasis']))
```

```
                precision    recall  f1-score   support

    Atelectasis       0.89      0.77      0.83      1826
No Atelectasis        0.12      0.25      0.16       222

      accuracy                           0.71      2048
     macro avg       0.50      0.51      0.49      2048
  weighted avg       0.81      0.71      0.76      2048
```



❖ Cardiomegaly

```
[ ] print('test accuracy = ',multi_disease_model.evaluate(test_X,test_y2, verbose=0)[1])
```

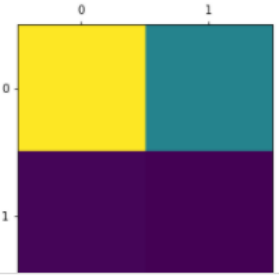
```
test accuracy =  0.97412109375
```

```
[ ] from sklearn.metrics import roc_auc_score
print('roc score = ',roc_auc_score(test_y2.astype(int), y_pred))
```

```
roc score =  0.513420343311108
```

```
[ ] from sklearn.metrics import classification_report, confusion_matrix
plt.matshow(confusion_matrix(test_y2, y_pred>3.6301106e-05))
print(classification_report(test_y2, y_pred>3.6301106e-05, target_names = ['Cardiomegaly', 'No Cardiomegaly']))
```

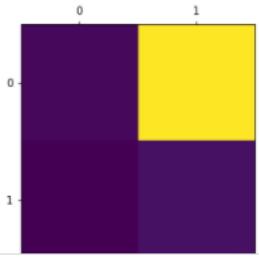
```
                precision    recall  f1-score   support

   Cardiomegaly       0.98      0.69      0.81      1995
No Cardiomegaly       0.03      0.34      0.05        53

      accuracy                           0.68      2048
     macro avg       0.50      0.51      0.43      2048
  weighted avg       0.95      0.68      0.79      2048
```



❖ Consolidation

```
[ ] print('test accuracy = ',multi_disease_model.evaluate(test_X,test_y2, verbose=0)[1])

    test accuracy =  0.9580078125

[ ] from sklearn.metrics import roc_auc_score
    print('roc score = ',roc_auc_score(test_y2.astype(int), y_pred))

    roc score =  0.5041248844321171

[ ] from sklearn.metrics import classification_report, confusion_matrix
    plt.matshow(confusion_matrix(test_y2, y_pred>3.52838e-05))
    print(classification_report(test_y2, y_pred>3.52838e-05, target_names = ['Consolidation', 'No Consolidation']))
```

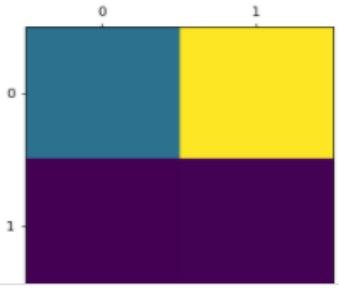
```
                  precision    recall  f1-score   support

   Consolidation       0.97      0.02      0.04      1962
No Consolidation       0.04      0.99      0.08        86

        accuracy                           0.06      2048
       macro avg       0.51      0.50      0.06      2048
    weighted avg       0.94      0.06      0.04      2048
```



❖ Edema

```
[ ] print('test accuracy = ',multi_disease_model.evaluate(test_X,test_y2, verbose=0)[1])

    test accuracy =  0.984375

[ ] from sklearn.metrics import roc_auc_score
    print('roc score = ',roc_auc_score(test_y2.astype(int), y_pred))

    roc score =  0.4650297619047619

[ ] from sklearn.metrics import classification_report, confusion_matrix
    plt.matshow(confusion_matrix(test_y2, y_pred>2.1467426e-05))
    print(classification_report(test_y2, y_pred>2.1467426e-05, target_names = ['Edema', 'No Edema']))
```

```
              precision    recall  f1-score   support

       Edema       0.98      0.27      0.43      2016
   No Edema        0.01      0.66      0.03        32

    accuracy                           0.28      2048
   macro avg       0.50      0.47      0.23      2048
weighted avg       0.97      0.28      0.42      2048
```



❖ Effusion

```
[ ] print('test accuracy = ',multi_disease_model.evaluate(test_X,test_y2, verbose=0)[1])
```

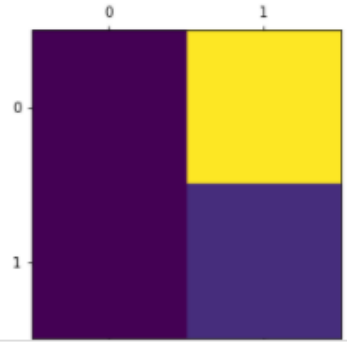
```
test accuracy =  0.88427734375
```

```
[ ] from sklearn.metrics import roc_auc_score
    print('roc score = ',roc_auc_score(test_y2.astype(int), y_pred))
```

```
roc score =  0.48639234681633803
```

```
[ ] from sklearn.metrics import classification_report, confusion_matrix
    plt.matshow(confusion_matrix(test_y2, y_pred>0.4883658))
    print(classification_report(test_y2, y_pred>0.4883658, target_names = ['Effusion', 'No Effusion']))
```

```
              precision    recall  f1-score   support

    Effusion       0.67      0.00      0.00      1811
 No Effusion       0.12      1.00      0.21       237

    accuracy                           0.12      2048
   macro avg       0.39      0.50      0.10      2048
weighted avg       0.60      0.12      0.03      2048
```



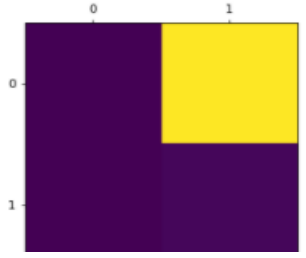
❖ Emphysema

```
[ ] print('test accuracy = ',multi_disease_model.evaluate(test_X,test_y2, verbose=0)[1])
```

```
test accuracy =  0.98095703125
```

```
[ ] from sklearn.metrics import classification_report, confusion_matrix
    plt.matshow(confusion_matrix(test_y2, y_pred>0.02157214))
    print(classification_report(test_y2, y_pred>0.02157214, target_names = ['Emphysema', 'No Emphysema']))
```

```
              precision    recall  f1-score   support

   Emphysema       0.00      0.00      0.00      2009
No Emphysema       0.02      1.00      0.04        39

    accuracy                           0.02      2048
   macro avg       0.01      0.50      0.02      2048
weighted avg       0.00      0.02      0.00      2048

/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetricWarning: Precision and
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetricWarning: Precision and
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetricWarning: Precision and
  _warn_prf(average, modifier, msg_start, len(result))
```



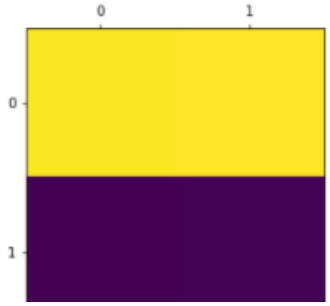
❖ Fibrosis

```
[ ] print('test accuracy = ',multi_disease_model.evaluate(test_X,test_y2, verbose=0)[1])

    test accuracy =  0.98583984375
```

```
[ ] from sklearn.metrics import roc_auc_score
    print('roc score = ',roc_auc_score(test_y2.astype(int), y_pred))

    roc score =  0.5417413878499087
```

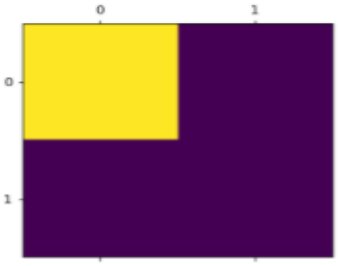
```
from sklearn.metrics import classification_report, confusion_matrix
plt.matshow(confusion_matrix(test_y2, y_pred>1.047299e-05))
print(classification_report(test_y2, y_pred>1.047299e-05, target_names = ['Fibrosis', 'No Fibrosis']))
```

```
              precision    recall  f1-score   support

    Fibrosis       0.99      0.50      0.66      2019
 No Fibrosis       0.02      0.59      0.03        29

    accuracy                           0.50      2048
   macro avg       0.50      0.54      0.35      2048
weighted avg       0.97      0.50      0.65      2048
```



❖ Hernia

```
[ ] from sklearn.metrics import classification_report, confusion_matrix
    plt.matshow(confusion_matrix(test_y2, pred_Y>0.0044266))
    print(classification_report(test_y2, pred_Y>0.0044266, target_names = ['No Hernia', 'Hernia']))

    /usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetricWa
      _warn_prf(average, modifier, msg_start, len(result))
    /usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetricWa
      _warn_prf(average, modifier, msg_start, len(result))
    /usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetricWa
      _warn_prf(average, modifier, msg_start, len(result))
                  precision    recall  f1-score   support

       No Hernia       1.00      1.00      1.00      2143
          Hernia       0.00      0.00      0.00         3

        accuracy                           1.00      2146
       macro avg       0.50      0.50      0.50      2146
    weighted avg       1.00      1.00      1.00      2146
```



❖ Infiltration

```
[ ] print('test accuracy = ',multi_disease_model.evaluate(test_X,test_y2, verbose=0)[1])
```

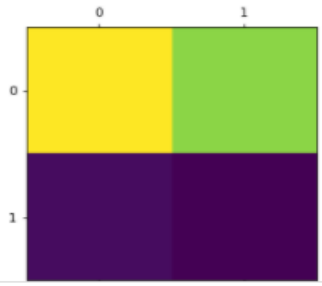
```
test accuracy =  0.82275390625
```

```
[ ] from sklearn.metrics import roc_auc_score
    print('roc score = ',roc_auc_score(test_y2.astype(int), y_pred))
```

```
roc score =  0.501625099116332
```

```
[ ] from sklearn.metrics import classification_report, confusion_matrix
    plt.matshow(confusion_matrix(test_y2, y_pred>0.15012267))
    print(classification_report(test_y2, y_pred>0.15012267, target_names = ['Infiltration', 'No Infiltration']))
```

```
                precision    recall  f1-score   support

   Infiltration       0.82      0.54      0.65      1685
No Infiltration       0.18      0.47      0.26       363

       accuracy                           0.52      2048
      macro avg       0.50      0.50      0.45      2048
   weighted avg       0.71      0.52      0.58      2048
```



❖ Mass

```
[ ] print('test accuracy = ',multi_disease_model.evaluate(test_X,test_y2, verbose=0)[1])
```

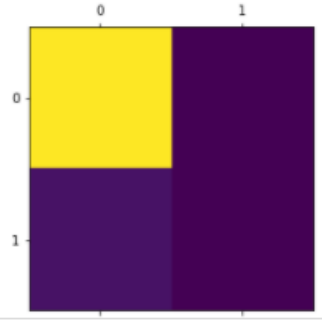
```
test accuracy =  0.951171875
```

```
[ ] from sklearn.metrics import roc_auc_score
    print('roc score = ',roc_auc_score(test_y2.astype(int), y_pred))
```

```
roc score =  0.5034599589322382
```

```
[ ] from sklearn.metrics import classification_report, confusion_matrix
    plt.matshow(confusion_matrix(test_y2, y_pred>1.614412e-05))
    print(classification_report(test_y2, y_pred>1.614412e-05, target_names = ['Mass', 'No Mass']))
```

```
              precision    recall  f1-score   support

        Mass       0.95      1.00      0.97      1948
     No Mass       0.14      0.01      0.02       100

    accuracy                           0.95      2048
   macro avg       0.55      0.50      0.50      2048
weighted avg       0.91      0.95      0.93      2048
```



❖ Nodule

```
[ ] print('test accuracy = ',multi_disease_model.evaluate(test_X,test_y2, verbose=0)[1])
```

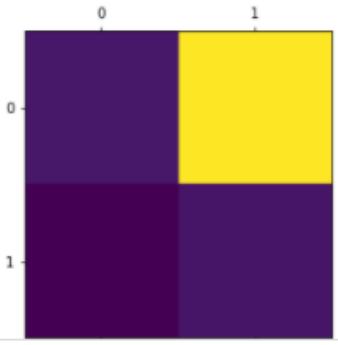
```
test accuracy =  0.94287109375
```

```
[ ] from sklearn.metrics import roc_auc_score
    print('roc score = ',roc_auc_score(test_y2.astype(int), y_pred))
```

```
roc score =  0.5016753199042168
```

```
[ ] from sklearn.metrics import classification_report, confusion_matrix
    plt.matshow(confusion_matrix(test_y2, y_pred>0.09439713))
    print(classification_report(test_y2, y_pred>0.09439713, target_names = ['Nodule', 'No Nodule']))
```

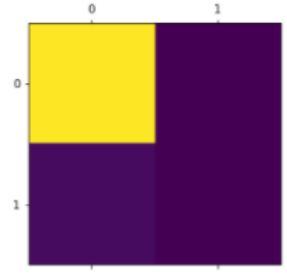
```
              precision    recall  f1-score   support

      Nodule       0.95      0.06      0.12      1931
   No Nodule       0.06      0.94      0.11       117

    accuracy                           0.11      2048
   macro avg       0.50      0.50      0.11      2048
weighted avg       0.89      0.11      0.12      2048
```



❖ Pleural-Thickening

```
[ ] from sklearn.metrics import classification_report, confusion_matrix
    plt.matshow(confusion_matrix(test_y2, y_pred>4.291682e-05))
    print(classification_report(test_y2, y_pred>4.291682e-05, target_names = ['Pleural_Thickening', 'No Pleural_Thickening']))
```

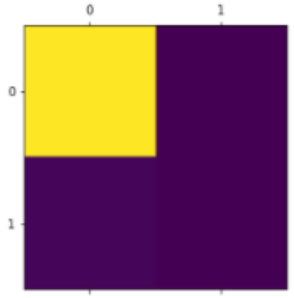
```
                      precision    recall  f1-score   support

   Pleural_Thickening       0.97      1.00      0.98      1987
No Pleural_Thickening       0.00      0.00      0.00        61

             accuracy                           0.97      2048
            macro avg       0.49      0.50      0.49      2048
         weighted avg       0.94      0.97      0.96      2048

/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetricWarning: Precision and F-sc
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetricWarning: Precision and F-sc
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetricWarning: Precision and F-sc
  _warn_prf(average, modifier, msg_start, len(result))
```



❖ Pneumonia

```
[ ] from sklearn.metrics import classification_report, confusion_matrix
    plt.matshow(confusion_matrix(test_y2, y_pred>0.00014827))
    print(classification_report(test_y2, y_pred>0.00014827, target_names = ['Pneumonia', 'No Pneumonia']))
```

```
              precision    recall  f1-score   support

   Pneumonia       0.99      1.00      0.99      2024
No Pneumonia       0.00      0.00      0.00        24

    accuracy                           0.99      2048
   macro avg       0.49      0.50      0.50      2048
weighted avg       0.98      0.99      0.98      2048

/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetricWarning: Pre
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetricWarning: Pre
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetricWarning: Pre
  _warn_prf(average, modifier, msg_start, len(result))
```



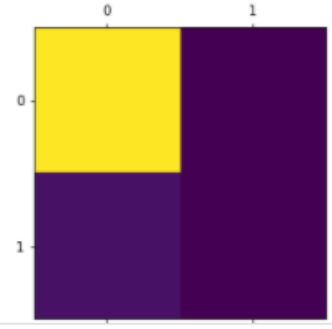
❖ Pneumothorax

```
print('test accuracy = ',multi_disease_model.evaluate(test_X,test_y2, verbose=0)[1])
```

```
test accuracy =  0.95263671875
```

```
[ ] from sklearn.metrics import classification_report, confusion_matrix
    plt.matshow(confusion_matrix(test_y2, y_pred>0.0847553))
    print(classification_report(test_y2, y_pred>0.0847553, target_names = ['Pneumothorax', 'No Pneumothorax']))
```

```
                precision    recall  f1-score   support

   Pneumothorax       0.95      1.00      0.98      1951
No Pneumothorax       0.00      0.00      0.00        97

       accuracy                           0.95      2048
      macro avg       0.48      0.50      0.49      2048
   weighted avg       0.91      0.95      0.93      2048

/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetricWarning: Precisio
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetricWarning: Precisio
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetricWarning: Precisio
  _warn_prf(average, modifier, msg_start, len(result))
```

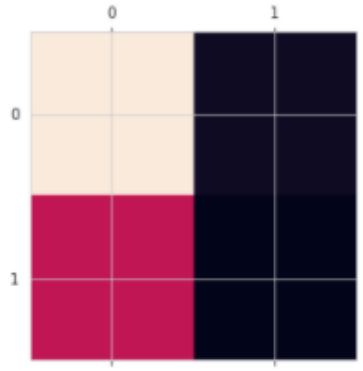


2. Undersampling

❖ Atelectasis

```
from sklearn.metrics import classification_report, confusion_matrix
plt.matshow(confusion_matrix(y_test, pred_Y>0.4813724))
print(classification_report(y_test, pred_Y>0.4813724, target_names = ['Healthy', 'Atelectasis']))
```

```
              precision    recall  f1-score   support

     Healthy       0.67      0.95      0.79      1458
 Atelectasis       0.23      0.03      0.06       688

    accuracy                           0.65      2146
   macro avg       0.45      0.49      0.42      2146
weighted avg       0.53      0.65      0.55      2146
```



❖ Cardiomegaly

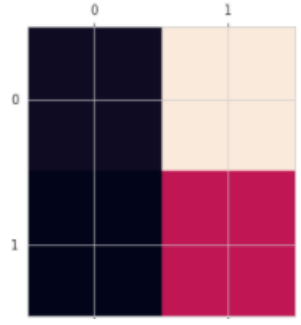
```
test accuracy =  0.6658900380134583
```

```
[51] from sklearn.metrics import roc_auc_score
     print('roc score = ',roc_auc_score(y_test.astype(int), pred_Y))
```

```
roc score =  0.49699734431135095
```

```
from sklearn.metrics import classification_report, confusion_matrix
plt.matshow(confusion_matrix(y_test, pred_Y>0.34658957))
print(classification_report(y_test, pred_Y>0.34658957, target_names = ['Healthy', 'Cardiomegaly']))
```

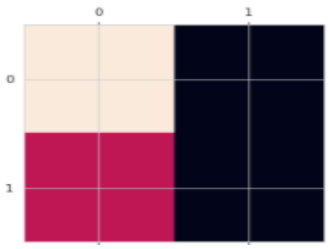
```
              precision    recall  f1-score   support

     Healthy       0.65      0.08      0.14      1429
Cardiomegaly       0.33      0.92      0.49       717

    accuracy                           0.36      2146
   macro avg       0.49      0.50      0.31      2146
weighted avg       0.54      0.36      0.26      2146
```



❖ Consolidation

```
[24] print('test accuracy = ',mobilenet_GAP_model.evaluate(X_test,y_test, verbose=0)[1])

     test accuracy =  0.677539587020874

[26] from sklearn.metrics import roc_auc_score
     print('roc score = ',roc_auc_score(y_test.astype(int), pred_Y))

     roc score =  0.5026854362293375

     from sklearn.metrics import classification_report, confusion_matrix
     plt.matshow(confusion_matrix(y_test, pred_Y>0.404212))
     print(classification_report(y_test, pred_Y>0.404212, target_names = ['Healthy', 'Consolidation']))
```

	precision	recall	f1-score	support
Healthy	0.68	0.99	0.81	1454
Consolidation	0.47	0.01	0.02	692
accuracy			0.68	2146
macro avg	0.57	0.50	0.41	2146
weighted avg	0.61	0.68	0.55	2146



❖ Edema

```
[24] print('test accuracy = ',mobilenet_GAP_model.evaluate(X_test,y_test, verbose=0)[1])

     test accuracy =  0.3314037621021271

[28] from sklearn.metrics import roc_auc_score
     print('roc score = ',roc_auc_score(y_test.astype(int), pred_Y))

     roc score =  0.5

[37] from sklearn.metrics import classification_report, confusion_matrix
     plt.matshow(confusion_matrix(y_test, pred_Y>0.5108716))
     print(classification_report(y_test, pred_Y>0.5108716, target_names = ['Healthy', 'Edema']))
```

	precision	recall	f1-score	support
Healthy	0.67	1.00	0.80	1386
Edema	0.33	1.00	0.50	687
accuracy			0.33	2073
macro avg	0.17	0.50	0.25	2073
weighted avg	0.11	0.33	0.16	2073

```
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetri
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetri
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetri
  _warn_prf(average, modifier, msg_start, len(result))
```

❖ Effusion

```
print('test accuracy = ',mobilenet_GAP_model.evaluate(X_test,y_test, verbose=0)[1])
```

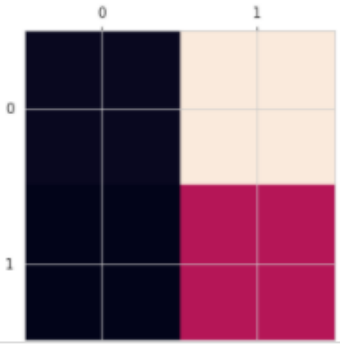
```
test accuracy =  0.67707359790802
```

```
[56] from sklearn.metrics import roc_auc_score
     print('roc score = ',roc_auc_score(y_test.astype(int), pred_Y))
```

```
roc score =  0.49245329114565173
```

```
from sklearn.metrics import classification_report, confusion_matrix
plt.matshow(confusion_matrix(y_test, pred_Y>0.3623708))
print(classification_report(y_test, pred_Y>0.3623708, target_names = ['Healthy', 'Effusion']))
```

```
              precision    recall  f1-score   support

     Healthy       0.62      0.06      0.11      1453
    Effusion       0.32      0.93      0.48       693

    accuracy                           0.34      2146
   macro avg       0.47      0.49      0.29      2146
weighted avg       0.53      0.34      0.23      2146
```



❖ Emphysema

```
[ ] print('test accuracy = ',mobilenet_GAP_model.evaluate(X_test,y_test, verbose=0)[1])
```

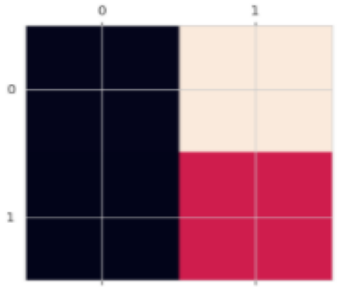
```
test accuracy =  0.661230206489563
```

```
from sklearn.metrics import roc_auc_score
print('roc score = ',roc_auc_score(y_test.astype(int), pred_Y))
```

```
roc score =  0.5025683080767691
```

```
[ ] from sklearn.metrics import classification_report, confusion_matrix
    plt.matshow(confusion_matrix(y_test, pred_Y>0.3948506))
    print(classification_report(y_test, pred_Y>0.3948506, target_names = ['No Emphysema', 'Emphysema']))
```

```
              precision    recall  f1-score   support

No Emphysema       0.76      0.01      0.03      1419
   Emphysema       0.34      0.99      0.51       727

    accuracy                           0.34      2146
   macro avg       0.55      0.50      0.27      2146
weighted avg       0.62      0.34      0.19      2146
```



❖ Fibrosis

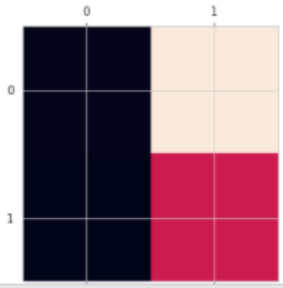
```
test accuracy =  0.664690375328064
```

```
[27] from sklearn.metrics import roc_auc_score
     print('roc score = ',roc_auc_score(y_test.astype(int), pred_Y))
```

```
roc score =  0.5030170508644206
```

```
from sklearn.metrics import classification_report, confusion_matrix
plt.matshow(confusion_matrix(y_test, pred_Y>0.42004278))
print(classification_report(y_test, pred_Y>0.42004278, target_names = ['Healthy', 'Fibrosis']))
```

```
              precision    recall  f1-score   support

     Healthy       0.76      0.02      0.03      1009
    Fibrosis       0.34      0.99      0.50       509

    accuracy                           0.34      1518
   macro avg       0.55      0.50      0.27      1518
weighted avg       0.62      0.34      0.19      1518
```



❖ Hernia

```
[23] print('test accuracy = ',mobilenet_GAP_model.evaluate(X_test,y_test, verbose=0)[1])
```

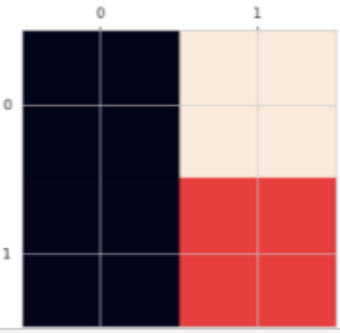
```
test accuracy =  0.6225489974021912
```

```
[24] from sklearn.metrics import roc_auc_score
     print('roc score = ',roc_auc_score(y_test.astype(int), pred_Y))
```

```
roc score =  0.4935064935064935
```

```
from sklearn.metrics import classification_report, confusion_matrix
plt.matshow(confusion_matrix(y_test, pred_Y>0.5302447))
print(classification_report(y_test, pred_Y>0.5302447, target_names = ['Healthy', 'Hernia']))
```

```
              precision    recall  f1-score   support

     Healthy       0.62      1.00      0.77       127

      Hernia       0.37      0.99      0.54        77

    accuracy                           0.37       204
   macro avg       0.19      0.49      0.27       204
weighted avg       0.14      0.37      0.20       204
```



❖ Infiltration

```
[23] print('test accuracy = ',mobilenet_GAP_model.evaluate(X_test,y_test, verbose=0)[1])
```

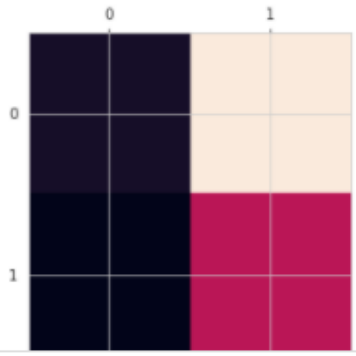
```
test accuracy =  0.6658900380134583
```

```
[25] from sklearn.metrics import roc_auc_score
     print('roc score = ',roc_auc_score(y_test.astype(int), pred_Y))
```

```
roc score =  0.4949662939332984
```

```
from sklearn.metrics import classification_report, confusion_matrix
plt.matshow(confusion_matrix(y_test, pred_Y>0.4872879))
print(classification_report(y_test, pred_Y>0.4872879, target_names = ['Healthy', 'Infiltration']))
```

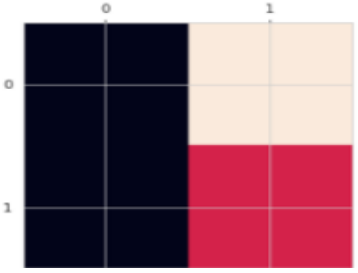
```
              precision    recall  f1-score   support

     Healthy       0.65      0.11      0.19      1429
Infiltration       0.33      0.88      0.48       717

    accuracy                           0.37      2146
   macro avg       0.49      0.49      0.34      2146
weighted avg       0.54      0.37      0.29      2146
```



❖ Mass

```
from sklearn.metrics import classification_report, confusion_matrix
plt.matshow(confusion_matrix(y_test, pred_Y>0.4442389))
print(classification_report(y_test, pred_Y>0.4442389, target_names = ['Healthy', 'Mass']))
```

```
              precision    recall  f1-score   support

     Healthy       0.65      1.00      0.79      1404

        Mass       0.35      1.00      0.51       742

    accuracy                           0.35      2146
   macro avg       0.17      0.50      0.26      2146
weighted avg       0.12      0.35      0.18      2146

/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetricWa
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetricWa
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetricWa
  _warn_prf(average, modifier, msg_start, len(result))
```



❖ Nodule

```
print('test accuracy = ',mobilenet_GAP_model.evaluate(X_test,y_test, verbose=0)[1])
```

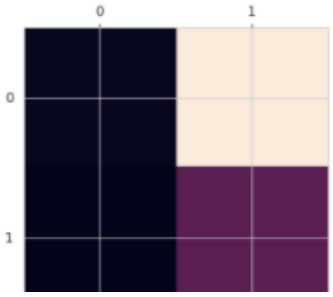
```
test accuracy =  0.8028891086578369
```

```
from sklearn.metrics import roc_auc_score
print('roc score = ',roc_auc_score(y_test.astype(int), pred_Y))
```

```
roc score =  0.495639580752138
```

```
from sklearn.metrics import classification_report, confusion_matrix
plt.matshow(confusion_matrix(y_test, pred_Y>0.20450142))
print(classification_report(y_test, pred_Y>0.20450142, target_names = ['Healthy', 'Nodule']))
```

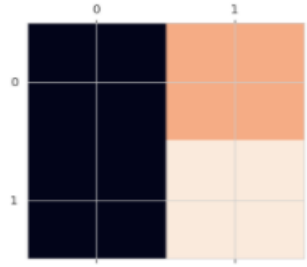
```
              precision    recall  f1-score   support

     Healthy       0.75      0.02      0.05      1723
      Nodule       0.20      0.97      0.33       423

    accuracy                           0.21      2146
   macro avg       0.47      0.50      0.19      2146
weighted avg       0.64      0.21      0.10      2146
```



❖ No Findings

```
from sklearn.metrics import classification_report, confusion_matrix
plt.matshow(confusion_matrix(y_test, pred_Y>0.4500000))
print(classification_report(y_test, pred_Y>0.45550000, target_names = ['Findings', 'No Findings']))
```

```
              precision    recall  f1-score   support

    Findings       0.00      0.00      0.00       979
 No Findings       0.54      1.00      0.70      1167

    accuracy                           0.54      2146
   macro avg       0.27      0.50      0.35      2146
weighted avg       0.30      0.54      0.38      2146

/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetricWarning
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetricWarning
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetricWarning
  _warn_prf(average, modifier, msg_start, len(result))
```



❖ Pleural-Thickening

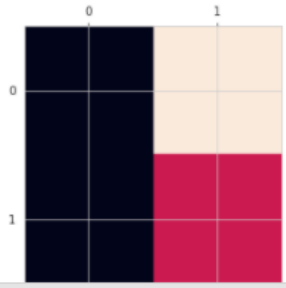
```
test accuracy =  0.659832239151001
```

```
[60] from sklearn.metrics import roc_auc_score
     print('roc score = ',roc_auc_score(y_test.astype(int), pred_Y))
```

```
roc score =  0.4936527745530532
```

```
from sklearn.metrics import classification_report, confusion_matrix
plt.matshow(confusion_matrix(y_test, pred_Y>0.4511414))
print(classification_report(y_test, pred_Y>0.4511414, target_names = ['Healthy', 'Pleural_Thickening']))
```

```
                   precision    recall  f1-score   support

          Healthy       0.47      0.01      0.02      1416
Pleural_Thickening       0.34      0.98      0.50       730

         accuracy                           0.34      2146
        macro avg       0.40      0.49      0.26      2146
     weighted avg       0.42      0.34      0.18      2146
```



❖ Pneumonia

```
[ ] print('test accuracy = ',mobilenet_GAP_model.evaluate(X_test,y_test, verbose=0)[1])
```

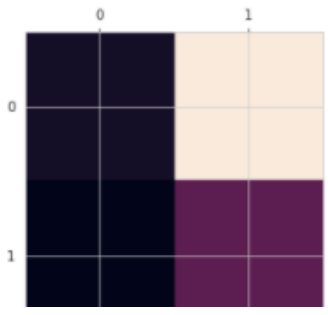
```
test accuracy =  0.7940354347229004
```

```
from sklearn.metrics import roc_auc_score
print('roc score = ',roc_auc_score(y_test.astype(int), pred_Y))sni
```

```
roc score =  0.4964987891147791
```

```
[ ] from sklearn.metrics import classification_report, confusion_matrix
    plt.matshow(confusion_matrix(y_test, pred_Y>0.19603807))
    print(classification_report(y_test, pred_Y>0.19603807, target_names = ['Healthy', 'Pneumonia']))
```

```
              precision    recall  f1-score   support

     Healthy       0.78      0.07      0.13      1704
   Pneumonia       0.20      0.92      0.33       442

    accuracy                           0.25      2146
   macro avg       0.49      0.50      0.23      2146
weighted avg       0.66      0.25      0.17      2146
```



❖ Pneumothorax

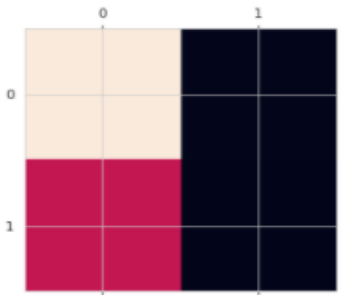
```
test accuracy =  0.6738117337226868
```

```
from sklearn.metrics import roc_auc_score
print('roc score = ',roc_auc_score(y_test.astype(int), pred_Y))
```

```
roc score =  0.49951294210630315
roc score =  0.49951294210630315
```

```
from sklearn.metrics import classification_report, confusion_matrix
plt.matshow(confusion_matrix(y_test, pred_Y>0.28713757))
print(classification_report(y_test, pred_Y>0.28713757, target_names = ['Healthy', 'Pneumothorax']))
```

```
              precision    recall  f1-score   support

     Healthy       0.67      0.99      0.80      1446
Pneumothorax       0.31      0.01      0.02       700

    accuracy                           0.67      2146
   macro avg       0.49      0.50      0.41      2146
weighted avg       0.56      0.67      0.55      2146
```



3. Oversampling

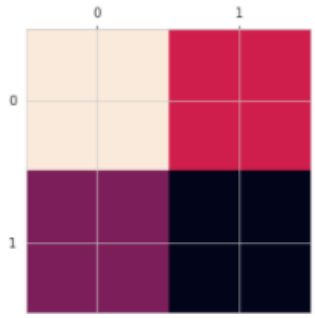
❖ Atelectasis

```
[65] print('test accuracy = ',mobilenet_GAP_model.evaluate(X_test,y_test, verbose=0)[1])

     test accuracy =  0.67940354347229
```

```
[66] from sklearn.metrics import roc_auc_score
     print('roc score = ',roc_auc_score(y_test.astype(int), pred_Y))

     roc score =  0.47941090853989216
```

```
from sklearn.metrics import classification_report, confusion_matrix
plt.matshow(confusion_matrix(y_test, pred_Y>0.21965626))
print(classification_report(y_test, pred_Y>0.21965626, target_names = ['Healthy', 'Atelectasis']))
```

```
              precision    recall  f1-score   support

     Healthy       0.66      0.61      0.63      1458
 Atelectasis       0.30      0.35      0.32       688

    accuracy                           0.52      2146
   macro avg       0.48      0.48      0.48      2146
weighted avg       0.55      0.52      0.53      2146
```



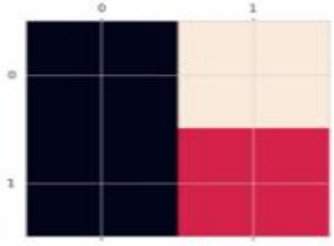
❖ Cardiomegaly

```
[25] print('test accuracy = ',mobilenet_GAP_model.evaluate(X_test,y_test, verbose=0)[1])

     test accuracy =  0.6547064185142517
```

```
[26] from sklearn.metrics import roc_auc_score
     print('roc score = ',roc_auc_score(y_test.astype(int), pred_Y))

     roc score =  0.5093914638773226
```

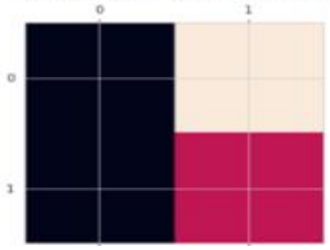
```
from sklearn.metrics import classification_report, confusion_matrix
plt.matshow(confusion_matrix(y_test, pred_Y>0.39474553))
print(classification_report(y_test, pred_Y>0.39474553, target_names = ['Healthy', 'Cardiomegaly']))
```

```
              precision    recall  f1-score   support

     Healthy       0.65      1.00      0.79      1405

Cardiomegaly       0.35      1.00      0.51       741

    accuracy                           0.35      2146
   macro avg       0.17      0.50      0.26      2146
weighted avg       0.12      0.35      0.18      2146

/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetricWarning:
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetricWarning:
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetricWarning:
  _warn_prf(average, modifier, msg_start, len(result))
```



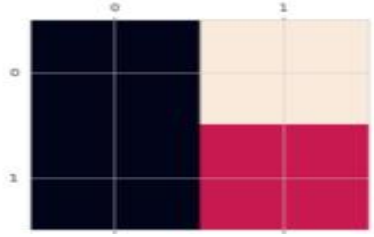
❖ Consolidation

```
[26] print('test accuracy = ',mobilenet_GAP_model.evaluate(X_test,y_test, verbose=0)[1])

     test accuracy =  0.677539587020874

     from sklearn.metrics import roc_auc_score
     print('roc score = ',roc_auc_score(y_test.astype(int), pred_Y))

     roc score =  0.5

     from sklearn.metrics import classification_report, confusion_matrix
     plt.matshow(confusion_matrix(y_test, pred_Y>0.48584896))
     print(classification_report(y_test, pred_Y>0.48584896, target_names = ['Healthy', 'Consolidation']))

                   precision    recall  f1-score   support

          Healthy       0.68      1.00      0.81      1454

    Consolidation       0.32      1.00      0.49       692

         accuracy                           0.32      2146
        macro avg       0.16      0.50      0.24      2146
     weighted avg       0.10      0.32      0.16      2146

     /usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetricWarning: Pr
       _warn_prf(average, modifier, msg_start, len(result))
     /usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetricWarning: Pr
       _warn_prf(average, modifier, msg_start, len(result))
     /usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetricWarning: Pr
       _warn_prf(average, modifier, msg_start, len(result))
```



❖ Edema

```
     print('test accuracy = ',mobilenet_GAP_model.evaluate(X_test,y_test, verbose=0)[1])

     test accuracy =  0.6685962080955505

[27] from sklearn.metrics import roc_auc_score
     print('roc score = ',roc_auc_score(y_test.astype(int), pred_Y))

     roc score =  0.49740543299495266

     from sklearn.metrics import classification_report, confusion_matrix
     plt.matshow(confusion_matrix(y_test, pred_Y>0.4963053))
     print(classification_report(y_test, pred_Y>0.4963053, target_names = ['Healthy', 'Edema']))

                   precision    recall  f1-score   support

          Healthy       0.67      1.00      0.80      1386

           Edema       0.33      1.00      0.50       687

         accuracy                           0.33      2073
        macro avg       0.17      0.50      0.25      2073
     weighted avg       0.11      0.33      0.16      2073

     /usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetri
       _warn_prf(average, modifier, msg_start, len(result))
     /usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetri
       _warn_prf(average, modifier, msg_start, len(result))
     /usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetri
       _warn_prf(average, modifier, msg_start, len(result))
```



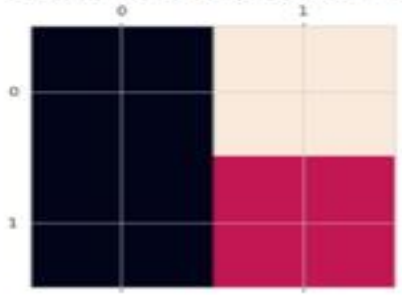
❖ Effusion

```
[26] print('test accuracy = ',mobilenet_GAP_model.evaluate(X_test,y_test, verbose=0)[1])

     test accuracy =  0.677087359790802
```

```
[ ] from sklearn.metrics import roc_auc_score
    print('roc score = ',roc_auc_score(y_test.astype(int), pred_Y))

    roc score =  0.5
```

```
from sklearn.metrics import classification_report, confusion_matrix
plt.matshow(confusion_matrix(y_test, pred_Y>0.35752255))
print(classification_report(y_test, pred_Y>0.35752255, target_names = ['Healthy', 'Effusion']))
```

```
              precision    recall  f1-score   support

     Healthy       0.68      1.00      0.81      1453

    Effusion       0.32      1.00      0.49       693

    accuracy                           0.32      2146
   macro avg       0.16      0.50      0.24      2146
weighted avg       0.10      0.32      0.16      2146

/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: Undefine
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: Undefine
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: Undefine
  _warn_prf(average, modifier, msg_start, len(result))
```



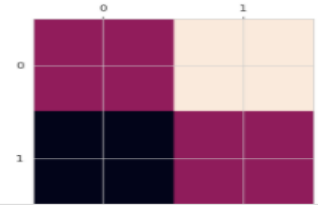
❖ Emphysema

```
[23] print('test accuracy = ',mobilenet_GAP_model.evaluate(X_test,y_test, verbose=0)[1])

     test accuracy =  0.661230206489563
```

```
[24] from sklearn.metrics import roc_auc_score
     print('roc score = ',roc_auc_score(y_test.astype(int), pred_Y))

     roc score =  0.5082996239869021
```

```
from sklearn.metrics import classification_report, confusion_matrix
plt.matshow(confusion_matrix(y_test, pred_Y>0.38926163))
print(classification_report(y_test, pred_Y>0.38926163, target_names = ['Healthy', 'Emphysema']))
```

```
              precision    recall  f1-score   support

     Healthy       0.67      0.35      0.46      1419
   Emphysema       0.34      0.67      0.45       727

    accuracy                           0.46      2146
   macro avg       0.51      0.51      0.46      2146
weighted avg       0.56      0.46      0.46      2146
```



❖ Fibrosis

```
print('test accuracy = ',mobilenet_GAP_model.evaluate(X_test,y_test, verbose=0)[1])
```

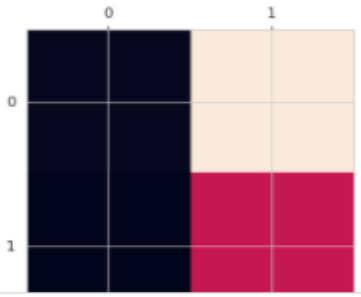
```
test accuracy =  0.664690375328064
```

```
[26] from sklearn.metrics import roc_auc_score
     print('roc score = ',roc_auc_score(y_test.astype(int), pred_Y))
```

```
roc score =  0.49528798767867194
```

```
from sklearn.metrics import classification_report, confusion_matrix
plt.matshow(confusion_matrix(y_test, pred_Y>0.33316213))
print(classification_report(y_test, pred_Y>0.33316213, target_names = ['Healthy', 'Fibrosis']))
```

```
              precision    recall  f1-score   support

     Healthy       0.62      0.04      0.08      1009
    Fibrosis       0.33      0.95      0.49       509

    accuracy                           0.35      1518
   macro avg       0.48      0.50      0.28      1518
weighted avg       0.52      0.35      0.22      1518
```



❖ Hernia

```
[24] print('test accuracy = ',mobilenet_GAP_model.evaluate(X_test,y_test, verbose=0)[1])
```

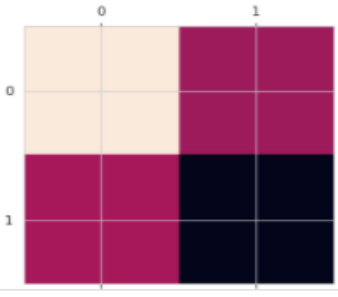
```
test accuracy =  0.6225489974021912
```

```
[25] from sklearn.metrics import roc_auc_score
     print('roc score = ',roc_auc_score(y_test.astype(int), pred_Y))
```

```
roc score =  0.492841803865426
```

```
from sklearn.metrics import classification_report, confusion_matrix
plt.matshow(confusion_matrix(y_test, pred_Y>0.40857682))
print(classification_report(y_test, pred_Y>0.40857682, target_names = ['Healthy', 'Hernia']))
```

```
              precision    recall  f1-score   support

     Healthy       0.62      0.62      0.62       127
      Hernia       0.37      0.36      0.37        77

    accuracy                           0.52       204
   macro avg       0.49      0.49      0.49       204
weighted avg       0.52      0.52      0.52       204
```



❖ Infiltration

```
print('test accuracy = ',mobilenet_GAP_model.evaluate(X_test,y_test, verbose=0)[1])
```

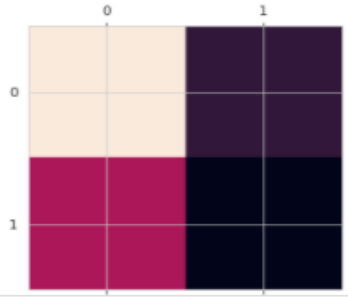
```
test accuracy =  0.6658900380134583
```

```
[27] from sklearn.metrics import roc_auc_score
     print('roc score = ',roc_auc_score(y_test.astype(int), pred_Y))
```

```
roc score =  0.5041923963954468
```

```
from sklearn.metrics import classification_report, confusion_matrix
plt.matshow(confusion_matrix(y_test, pred_Y>0.41862535))
print(classification_report(y_test, pred_Y>0.41862535, target_names = ['Healthy', 'Infiltration']))
```

```
              precision    recall  f1-score   support

     Healthy       0.67      0.80      0.73      1429
Infiltration       0.34      0.20      0.26       717

    accuracy                           0.60      2146
   macro avg       0.51      0.50      0.49      2146
weighted avg       0.56      0.60      0.57      2146
```



❖ Mass

```
[49] print('test accuracy = ',mobilenet_GAP_model.evaluate(X_test,y_test, verbose=0)[1])
```

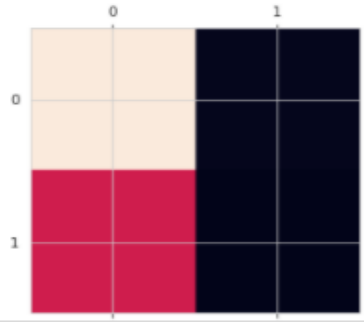
```
test accuracy =  0.6542404294013977
```

```
[47] from sklearn.metrics import roc_auc_score
     print('roc score = ',roc_auc_score(y_test.astype(int), pred_Y))
```

```
roc score =  0.5024477618817242
```

```
from sklearn.metrics import classification_report, confusion_matrix
plt.matshow(confusion_matrix(y_test, pred_Y>0.3194025))
print(classification_report(y_test, pred_Y>0.3194025, target_names = ['Healthy', 'Mass']))
```

```
              precision    recall  f1-score   support

     Healthy       0.66      0.97      0.78      1404
        Mass       0.38      0.04      0.07       742

    accuracy                           0.65      2146
   macro avg       0.52      0.50      0.43      2146
weighted avg       0.56      0.65      0.54      2146
```

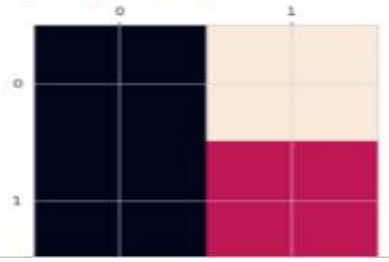


❖ Nodule

```
[66] print('test accuracy = ',mobilenet_GAP_model.evaluate(X_test,y_test, verbose=0)[1])

     test accuracy =  0.680335521697998

[67] from sklearn.metrics import roc_auc_score
     print('roc score = ',roc_auc_score(y_test.astype(int), pred_Y))

     roc score =  0.5

     from sklearn.metrics import classification_report, confusion_matrix
     plt.matshow(confusion_matrix(y_test, pred_Y>0.5))
     print(classification_report(y_test, pred_Y>0.5, target_names = ['Healthy', 'Nodule']))
```

	precision	recall	f1-score	support
Healthy	0.68	1.00	0.81	1460
Nodule	0.32	1.00	0.48	686
accuracy			0.32	2146
macro avg	0.16	0.50	0.24	2146
weighted avg	0.10	0.32	0.15	2146

```
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: U
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: U
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: U
  _warn_prf(average, modifier, msg_start, len(result))
```

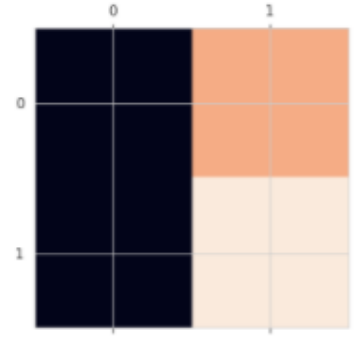


❖ No Findings

```
[ ] from sklearn.metrics import classification_report, confusion_matrix
    plt.matshow(confusion_matrix(y_test, pred_Y>0.5343945))
    print(classification_report(y_test, pred_Y>0.5343945, target_names = ['Findings', 'No Findings']))
```

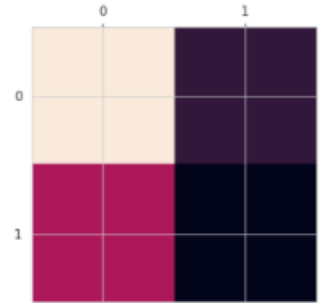
	precision	recall	f1-score	support
Findings	0.00	0.00	0.00	979
No Findings	0.54	1.00	0.70	1167
accuracy			0.54	2146
macro avg	0.27	0.50	0.35	2146
weighted avg	0.30	0.54	0.38	2146

```
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetricWarning
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetricWarning
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: UndefinedMetricWarning
  _warn_prf(average, modifier, msg_start, len(result))
```



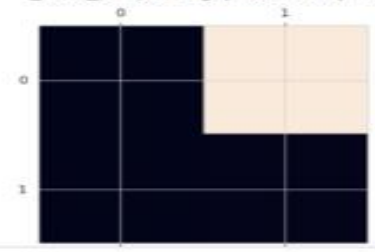
❖ Pleural-Thickening

```
[26] print('test accuracy = ',mobilenet_GAP_model.evaluate(X_test,y_test, verbose=0)[1])

     test accuracy =  0.6658900380134583

[27] from sklearn.metrics import roc_auc_score
     print('roc score = ',roc_auc_score(y_test.astype(int), pred_Y))

     roc score =  0.5041923963954468

     from sklearn.metrics import classification_report, confusion_matrix
     plt.matshow(confusion_matrix(y_test, pred_Y>0.41862535))
     print(classification_report(y_test, pred_Y>0.41862535, target_names = ['Healthy', 'Pleural_Thickening']))

                   precision    recall  f1-score   support

          Healthy       0.67      0.80      0.73      1429
     Infiltration       0.34      0.20      0.26       717

         accuracy                           0.60      2146
        macro avg       0.51      0.50      0.49      2146
     weighted avg       0.56      0.60      0.57      2146
```



❖ Pneumonia

```
[42] print('test accuracy = ',mobilenet_GAP_model.evaluate(X_test,y_test, verbose=0)[1])

     test accuracy =  0.9888163805007935

[43] from sklearn.metrics import roc_auc_score
     print('roc score = ',roc_auc_score(y_test.astype(int), pred_Y))

     roc score =  0.5

     from sklearn.metrics import classification_report, confusion_matrix
     plt.matshow(confusion_matrix(y_test, pred_Y>0.02324778))
     print(classification_report(y_test, pred_Y>0.02324778, target_names = ['Healthy', 'Pneumonia']))

                   precision    recall  f1-score   support

          Healthy       0.99      1.00      0.99      2122

        Pneumonia       0.01      1.00      0.02        24

         accuracy                           0.01      2146
        macro avg       0.01      0.50      0.01      2146
     weighted avg       0.00      0.01      0.00      2146

/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: Undefine
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: Undefine
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1308: Undefine
  _warn_prf(average, modifier, msg_start, len(result))
```



❖ Pneumothorax

```
[27] print('test accuracy = ',mobilenet_GAP_model.evaluate(X_test,y_test, verbose=0)[1])

     test accuracy =  0.6738117337226868
```

```
[28] from sklearn.metrics import roc_auc_score
     print('roc score = ',roc_auc_score(y_test.astype(int), pred_Y))

     roc score =  0.4999772772179411
```

```
from sklearn.metrics import classification_report, confusion_matrix
plt.matshow(confusion_matrix(y_test, pred_Y>0.59800578))
print(classification_report(y_test, pred_Y>0.59800578, target_names = ['Healthy', 'Pneumothorax']))
```

```
                precision    recall  f1-score   support

       Healthy       0.67      1.00      0.81      1446

Pneumothorax       0.33      1.00      0.49       700

    accuracy                           0.33      2146
   macro avg       0.50      0.50      0.25      2146
weighted avg       0.56      0.33      0.16      2146
```