mre_measure_cutoff

May 18, 2022

1 Liver Stiffness Cutoff

Author: Nana K. Owusu

Purpose: To determine the cutoff value for shear stiffness in determining the most accurate fibrosis stage.

1.1 Initial libraries

[1]:
```
from IPython.core.display import display, HTML
display(HTML("<style>.container{width:100% !important;}</style>"))
import numpy as np
import pandas as pd
pd.set_option("display.max_rows", None, "display.max_columns", None)
import matplotlib.pyplot as plt
from sklearn.metrics import confusion_matrix
# from timeit import default_timer as timer
%matplotlib inline
```

```
<IPython.core.display.HTML object>
```

1.2 Getting and Cleaning the Data

[2]:
```
#RAW DATA IMPORT
data=pd.read_csv('../nafld_cohort_60Hz.csv')

#RENAMING COLUMNS
data.rename({'Fibrosis-statistic':'floorFib'},axis=1, inplace=True)
# data=data.rename(columns=dict(zip(data.columns,new_names)))

#ADDING TARGET COLUMNS
data['Fib_gt0'] = data['floorFib'].apply(lambda x: int(x>0))
data['Fib_gt1'] = data['floorFib'].apply(lambda x: int(x>1))
data['Fib_gt2'] = data['floorFib'].apply(lambda x: int(x>2))
data['Fib_gt3'] = data['floorFib'].apply(lambda x: int(x>3))

#SELECTING RELEVANT COLUMNS
data=data[['ID','Age','Gender','NASH+F2','R2*','mu_SS','mu_AT','mu_VS',
```

```
            'PDFF','floorFib','Fib_gt0','Fib_gt1','Fib_gt2','Fib_gt3']]

#REMOVING MISSING VALUES
data.dropna(inplace=True)

#RESETTING INDEX
data.reset_index(drop=True, inplace=True)
```

[ ]:
```
#SUMMARY
print('\nDATA TYPES:\n-----------')
print(data.dtypes)
print('\nMISSING VALUES:\n---------------')
print(data.isna().sum())
print('\nDIMENSIONS:\n---------------')
print(data.shape)
data.head()
```

1.2.1 Function for calculating the cutoff accuracy

[3]:
```
def youden_cutoff(y_true, y_pred):

    # Calculate the confusion matrix from
    # the predicted and true values.
    tru_neg, _, _, tru_pos = \
    confusion_matrix(y_true, y_pred, normalize=None).ravel()

    # Derive the negative and positive
    # as well as the sensitivity and specificity.
    pos = y_true.sum()
    neg = len(y_true) - pos
    sens = 1-tru_pos/pos
    spec = 1-tru_neg/neg
    youden_idx = sens + spec - 1

    # balanced accuracy
    bal_acc = 0.5 * (youden_idx + 1)

    return (sens, spec, youden_idx, bal_acc)
```

1.3 Generate Balanced Accuracy

[4]:
```
mre_cutoff = np.linspace(0, 6, num=64)
cutoffs = pd.DataFrame(index=mre_cutoff, columns=['sens','spec','Youden Index',
                                                  'Balanced Accuracy'])
```

[13]:
```
for cutoff in mre_cutoff:

    flsneg_trupos = pd.Series(data['mu_SS']<=cutoff).astype('int16')
    sens, spec, youden_idx, ba = \
    youden_cutoff(data['Fib_gt0'], flsneg_trupos)

    cutoffs.loc[cutoff,'sens'] = sens
    cutoffs.loc[cutoff,'spec'] = spec
    cutoffs.loc[cutoff,'Youden Index'] = youden_idx
    cutoffs.loc[cutoff,'Balanced Accuracy'] = ba
```

[ ]:
```
cutoffs
```

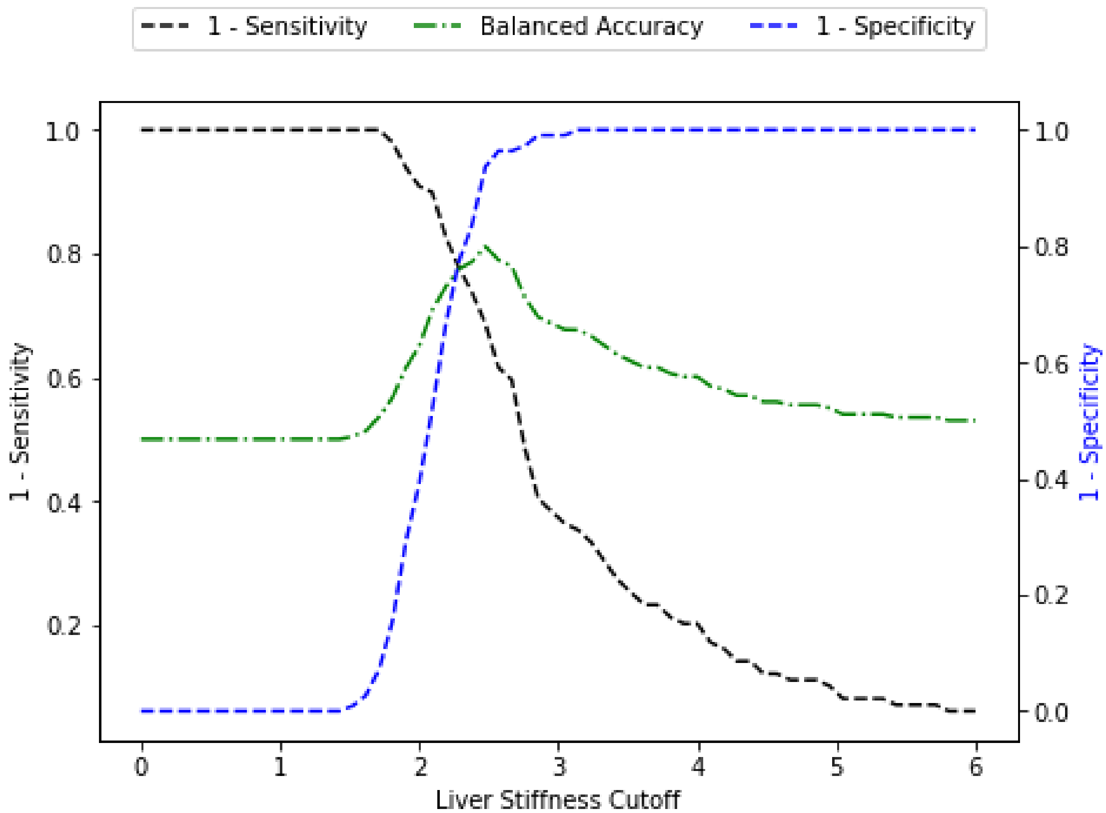
1.4 Plotting Cutoffs

[14]:
```
ss_cutoff_fig, ax1 = plt.subplots(figsize=(7,5))
ax2 = ax1.twinx()

ax1.plot(mre_cutoff, cutoffs['sens'], '--k', label='1 - Sensitivity')
ax2.plot(mre_cutoff, cutoffs['spec'], '--b', label='1 - Specificity')
ax1.plot(mre_cutoff, cutoffs['Balanced Accuracy'], '-.g', label='Balanced␣
 ↪Accuracy')
ax1.set_xlabel('Liver Stiffness Cutoff')
ax1.set_ylabel('1 - Sensitivity', color='k')
ax2.set_ylabel('1 - Specificity', color='b')

ss_cutoff_fig.legend(ncol=3, loc='upper center')
```

[14]:
```
<matplotlib.legend.Legend at 0x27e20483490>
```



1.4.1 Calculating the maximum accuracy

[18]:
```
mre_cutoff[cutoffs['Balanced Accuracy'].values.argmax()]
```

[18]: 2.4761904761904763