plot_roc

June 11, 2021

[1]:

```
%matplotlib inline
import os
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

sns.set_palette('Dark2')
```

[2]:

```
csv_path = '../outputs'
bal_basename = '_roc_details_balanced.csv'
unbal_basename = '_roc_details_unbalanced.csv'
```

[3]:

```
cnn_bal = pd.read_csv(os.path.join(csv_path, 'CNNModel' + bal_basename))
cnn_unbal = pd.read_csv(os.path.join(csv_path, 'CNNModel' + unbal_basename))
```

[4]:

```
feature_bal = pd.read_csv(os.path.join(csv_path, 'FeatureModel' + bal_basename))
feature_unbal = pd.read_csv(os.path.join(csv_path, 'FeatureModel' +␣
 ↪unbal_basename))
```

[5]:

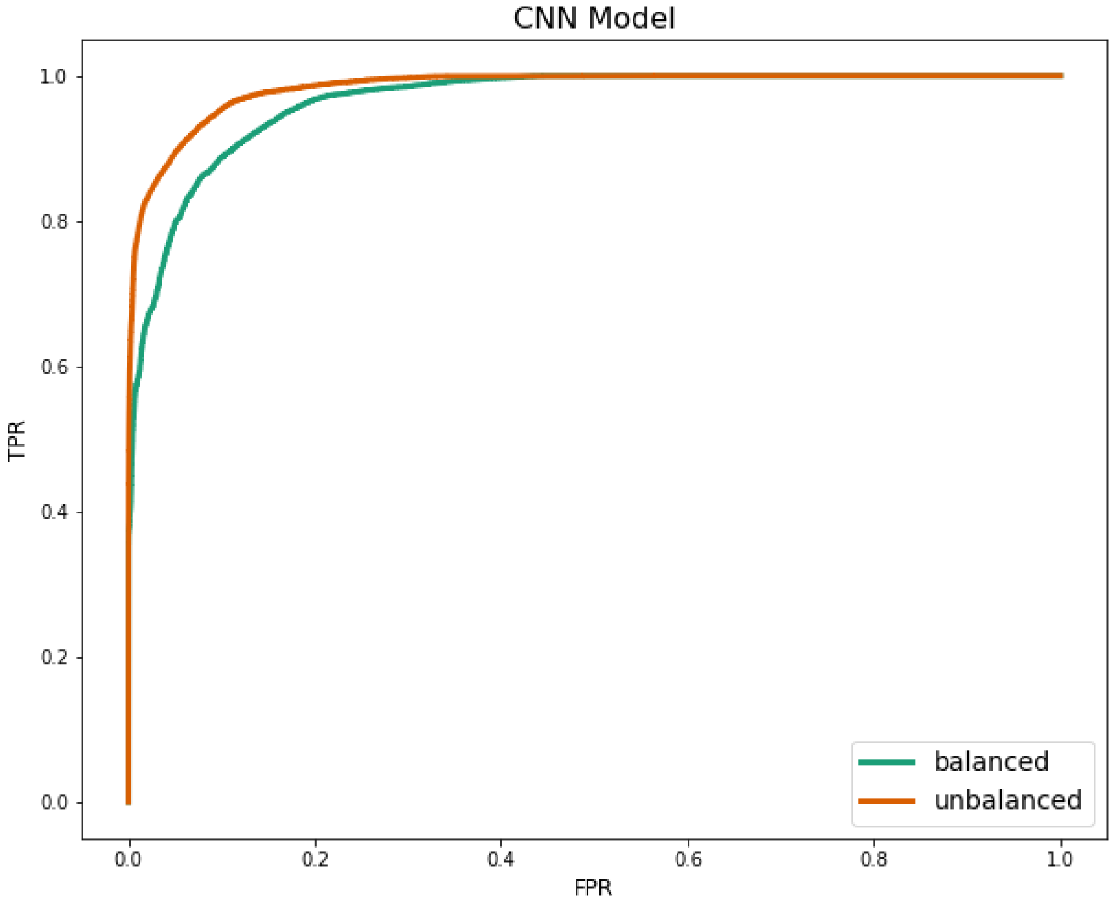
```
elm_bal = pd.read_csv(os.path.join(csv_path, 'StackedELMModel' + bal_basename))
elm_unbal = pd.read_csv(os.path.join(csv_path, 'StackedELMModel' +␣
 ↪unbal_basename))
```

[6]:

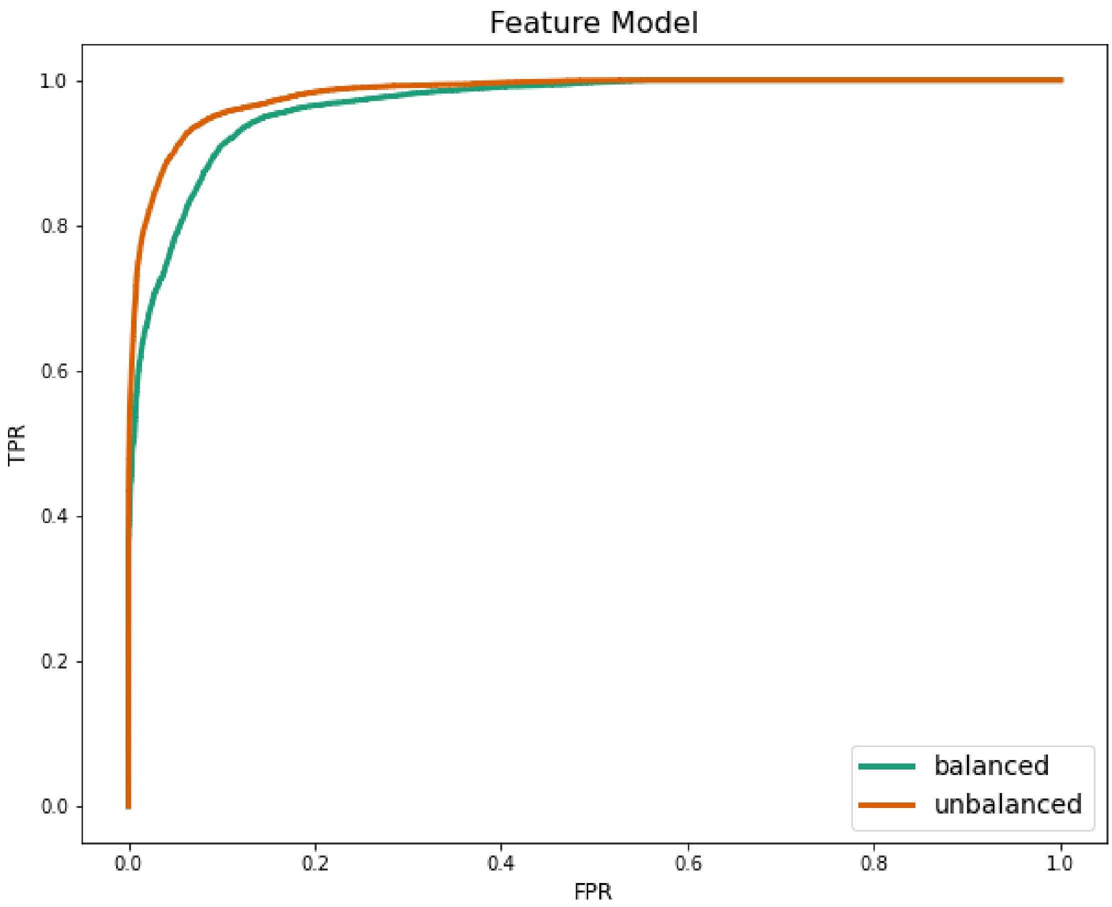
```
def show_roc(df1, df2, title=''):
    plt.figure(figsize=(10,8))
    plt.plot(df1['fpr'], df1['tpr'], lw=3, label='balanced')
    plt.plot(df2['fpr'], df2['tpr'], lw=3, label='unbalanced')
    plt.xlabel('FPR', fontsize=12)
    plt.ylabel('TPR', fontsize=12)
    plt.legend(fontsize=14)
    plt.title(title, fontsize=16)
    plt.show()
```

[7]:

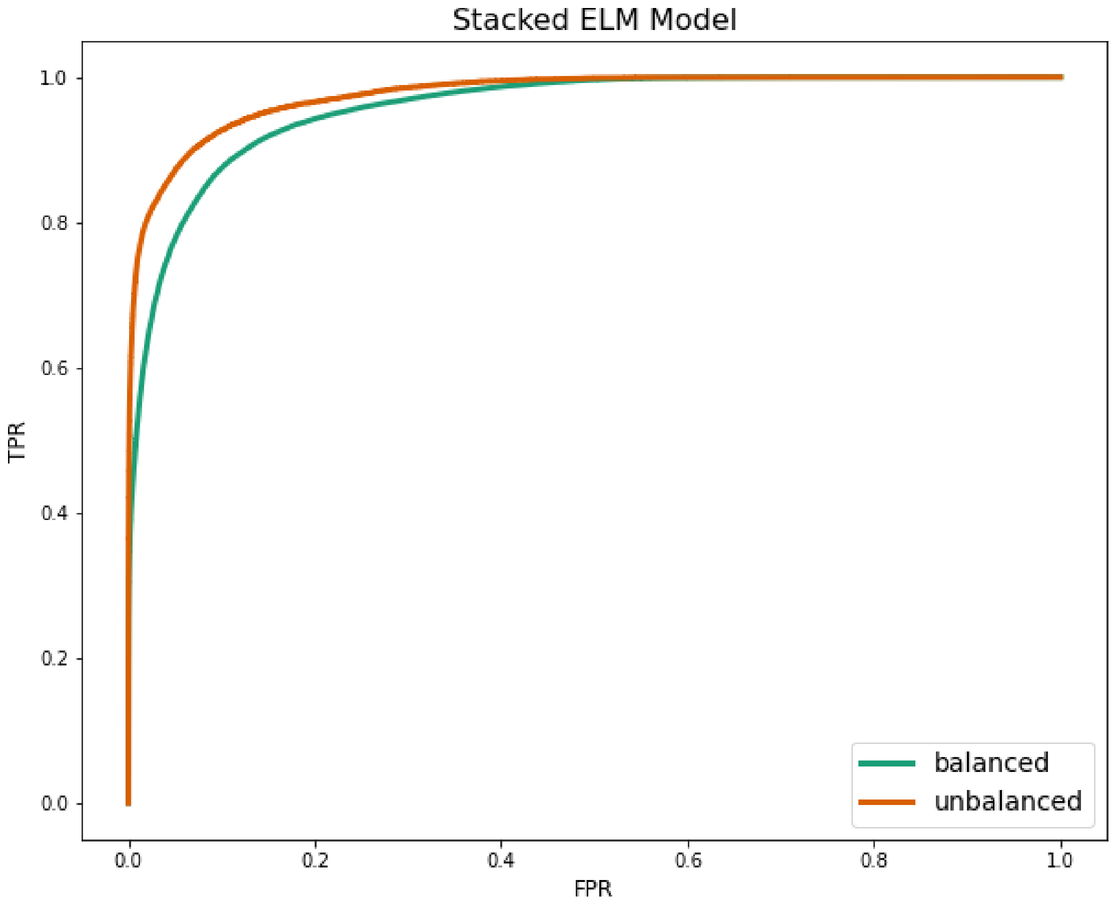
```
show_roc(cnn_bal, cnn_unbal, title='CNN Model')
```



```
[8]: show_roc(feature_bal, feature_unbal, title='Feature Model')
```

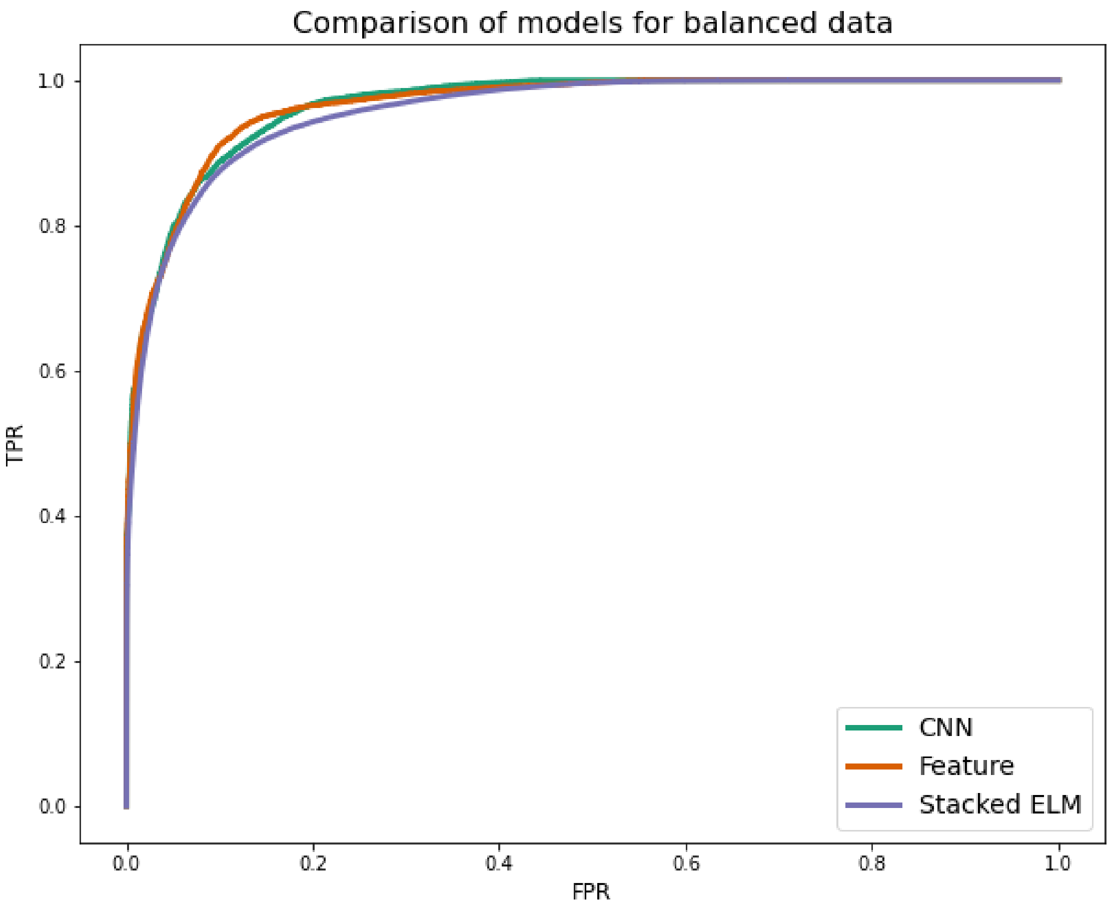


[9]:
```
show_roc(elm_bal, elm_unbal, title='Stacked ELM Model')
```



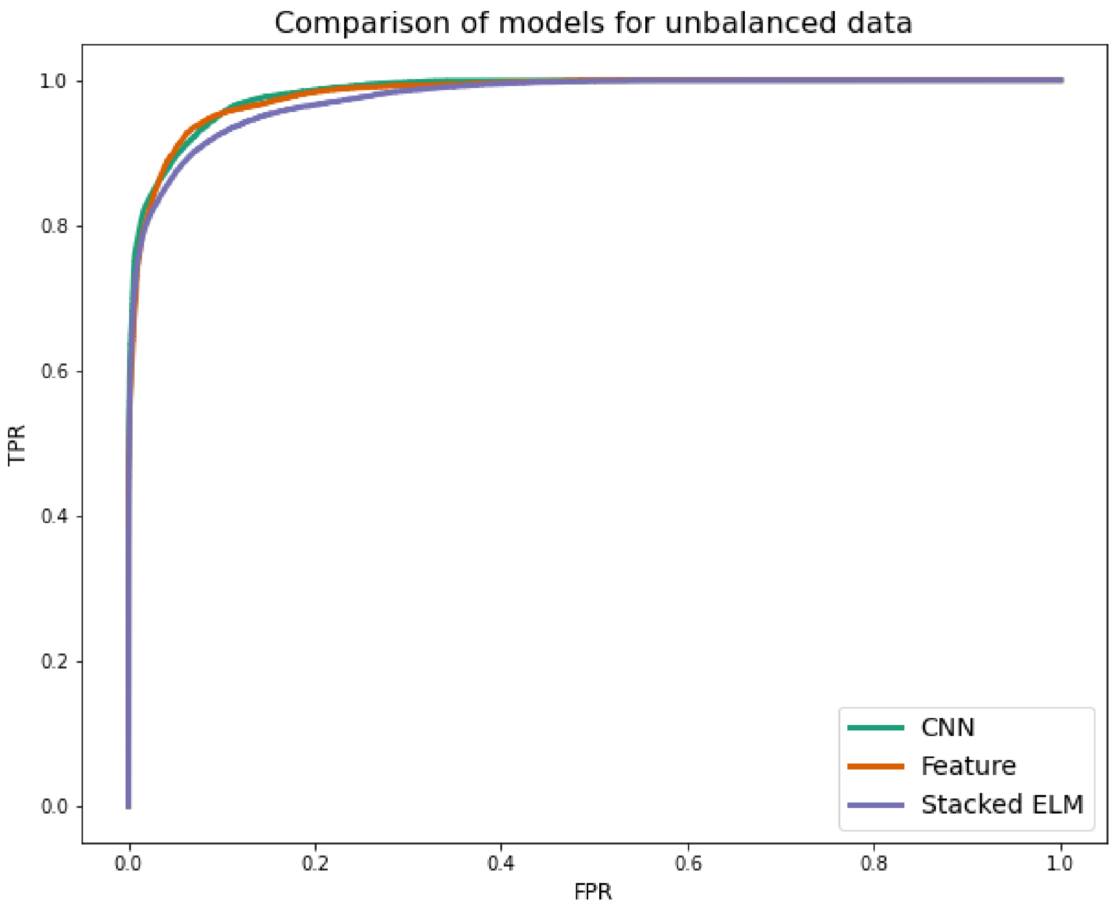
[11]:

```
plt.figure(figsize=(10,8))
plt.plot(cnn_bal['fpr'], cnn_bal['tpr'], lw=3, label='CNN')
plt.plot(feature_bal['fpr'], feature_bal['tpr'], lw=3, label='Feature')
plt.plot(elm_bal['fpr'], elm_bal['tpr'], lw=3, label='Stacked ELM')
plt.xlabel('FPR', fontsize=12)
plt.ylabel('TPR', fontsize=12)
plt.legend(fontsize=14)
plt.title('Comparison of models for balanced data', fontsize=16)
plt.show()
```



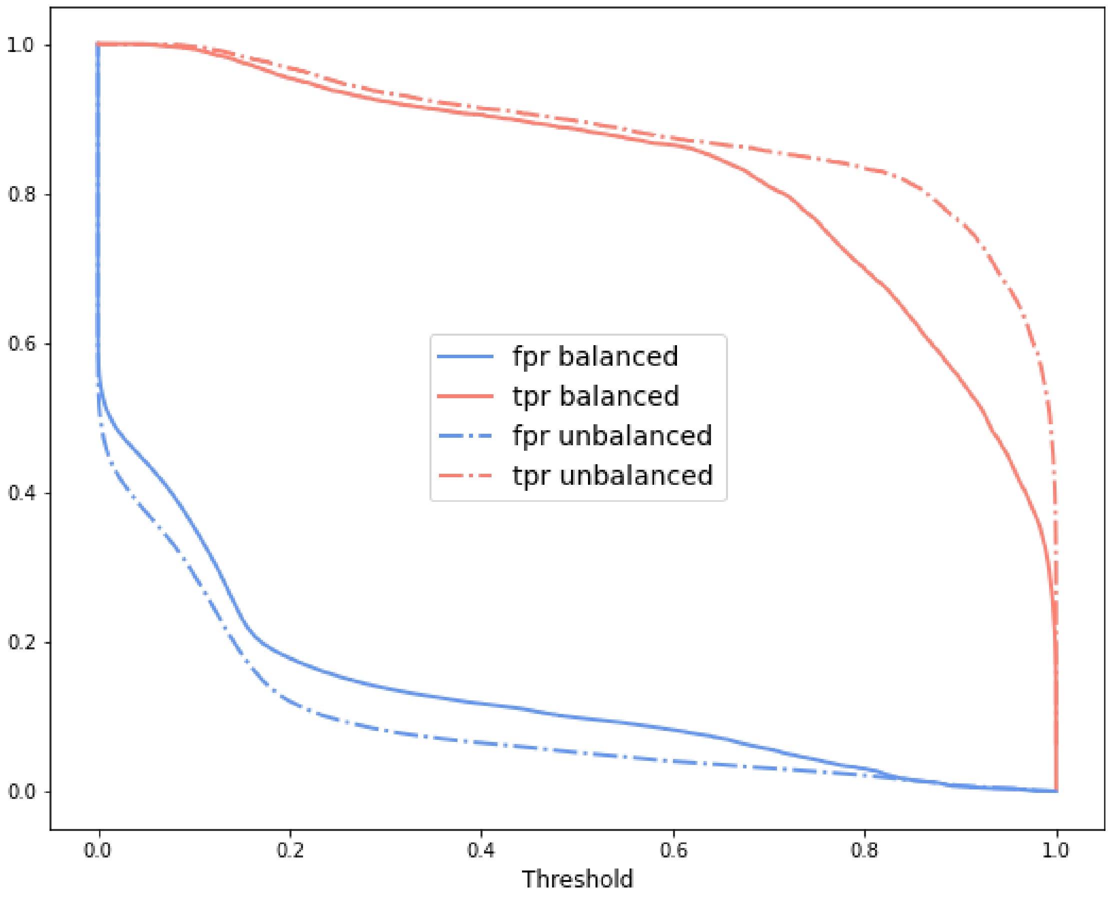
[12]:

```
plt.figure(figsize=(10,8))
plt.plot(cnn_unbal['fpr'], cnn_unbal['tpr'], lw=3, label='CNN')
plt.plot(feature_unbal['fpr'], feature_unbal['tpr'], lw=3, label='Feature')
plt.plot(elm_unbal['fpr'], elm_unbal['tpr'], lw=3, label='Stacked ELM')
plt.xlabel('FPR', fontsize=12)
plt.ylabel('TPR', fontsize=12)
plt.legend(fontsize=14)
plt.title('Comparison of models for unbalanced data', fontsize=16)
plt.show()
```

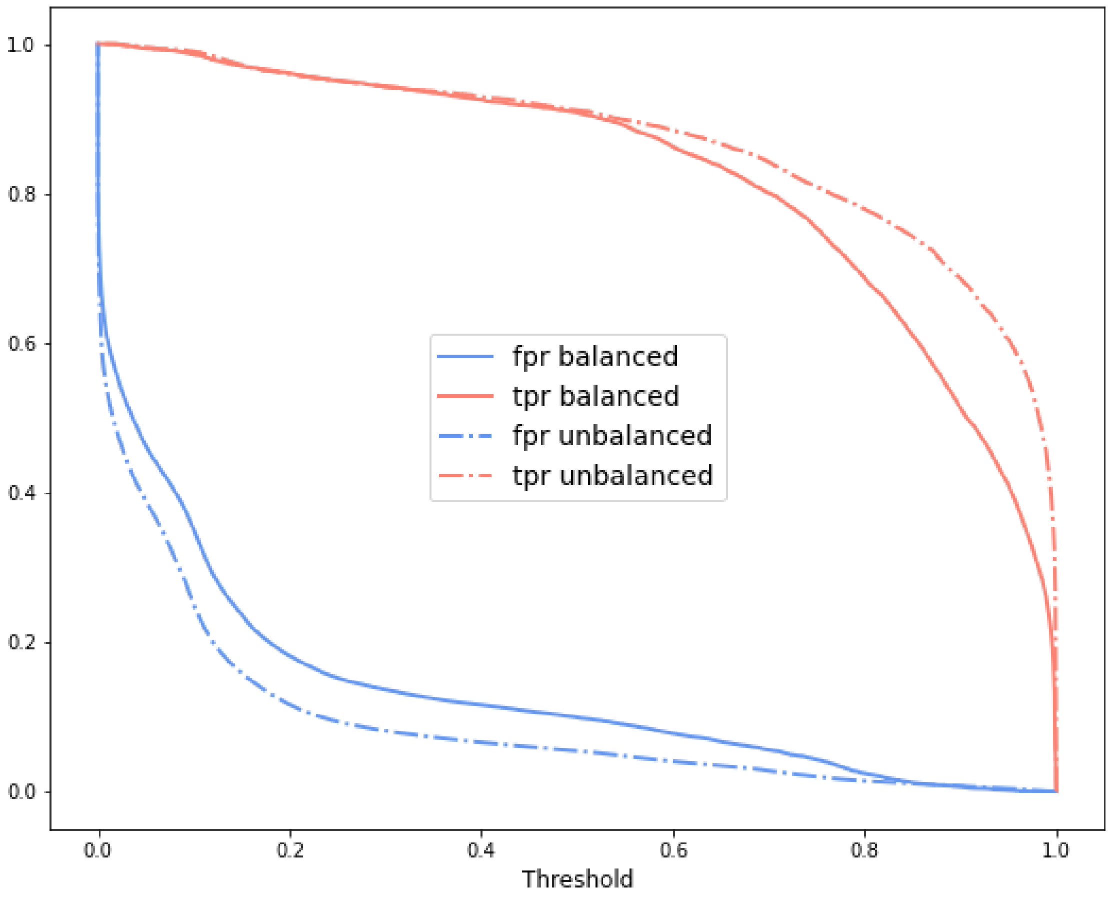


```
[34]: def show_thresholds(df1, df2):
          plt.figure(figsize=(10,8))
          plt.plot(df1['threshold'].loc[1:], df1['fpr'].loc[1:], '-', lw=2,␣
       ↪c='cornflowerblue', label='fpr balanced')
          plt.plot(df1['threshold'].loc[1:], df1['tpr'].loc[1:], '-', lw=2,␣
       ↪c='salmon', label='tpr balanced')
          plt.plot(df2['threshold'].loc[1:], df2['fpr'].loc[1:], '-.', lw=2,␣
       ↪c='cornflowerblue', label='fpr unbalanced')
          plt.plot(df2['threshold'].loc[1:], df2['tpr'].loc[1:], '-.', lw=2,␣
       ↪c='salmon', label='tpr unbalanced')
          plt.xlabel('Threshold', fontsize=12)
          plt.legend(fontsize=14)
          # plt.title(title, fontsize=16)
          plt.show()
```

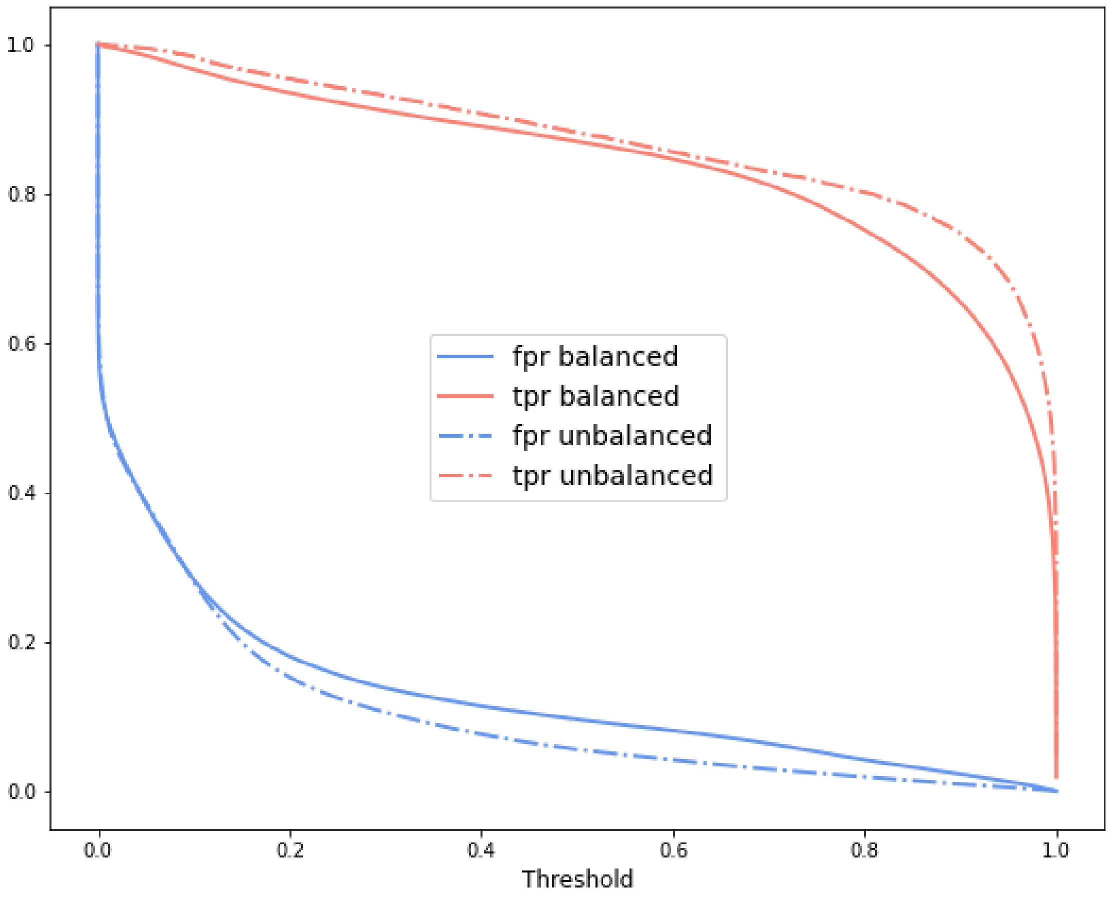
```
[35]: show_thresholds(cnn_bal, cnn_unbal)
```



[36]:
```
show_thresholds(feature_bal, feature_unbal)
```



[37]:
```
show_thresholds(elm_bal, elm_unbal)
```


1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
Threshold
fpr balanced
tpr balanced
fpr unbalanced
tpr unbalanced

```
[ ]:
```