Naivebayes

December 13, 2019

[1]:
```
from sklearn.model_selection import train_test_split, cross_val_score, KFold,␣
 ↪StratifiedKFold
# machine learning
from sklearn.naive_bayes import GaussianNB
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
from sklearn import metrics
import numpy as np
```

[2]:
```
train_df = pd.read_csv('train.csv')
test_df = pd.read_csv('test.csv')
```

[3]:
```
act_test_df = pd.read_csv('act_test.csv', dtype={'people_id': np.str,␣
 ↪'activity_id': np.str},
                          parse_dates=['date'])
```

[4]:
```
test_id = act_test_df.activity_id
```

[5]:
```
X_train = train_df.drop(['outcome'], axis=1)
Y_train = train_df['outcome']
```

[6]:
```
# train, validation set split
```

[7]:
```
x_train, x_val, y_train, y_val = train_test_split(X_train, Y_train, test_size =␣
 ↪0.5, random_state=1)
x_train.shape, x_val.shape, y_train.shape, y_val.shape
```

[7]:
```
((1098645, 59), (1098646, 59), (1098645,), (1098646,))
```

[8]:
```
gaussian = GaussianNB()
gaussian.fit(x_train, y_train)
acc_gaussian = round(gaussian.score(x_val,y_val) * 100, 2)
acc_gaussian
```

[8]:
```
68.64
```

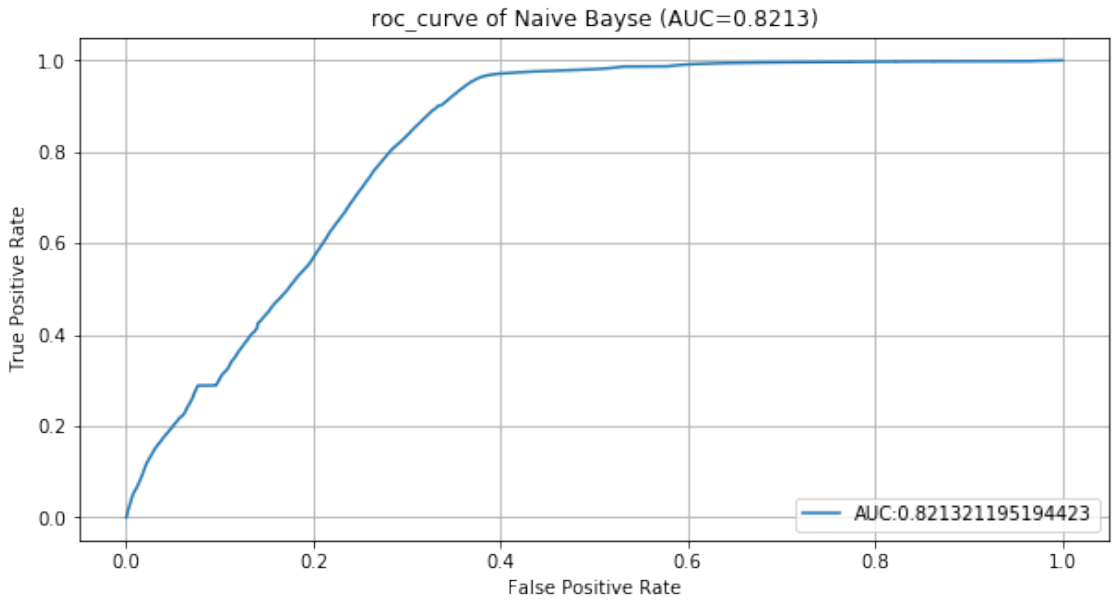
[9]:
```
gaussian.score(x_val, y_val)
```

[9]:
```
0.6864458615422985
```

[10]:
```
gaussian_predictions = gaussian.predict_proba(x_val)[:,1]
fpr, tpr, thresholds = metrics.roc_curve(y_val,gaussian_predictions)
gaussian_roc = pd.DataFrame()
gaussian_roc['fpr'] = fpr
gaussian_roc['threshold'] = thresholds
auc = metrics.roc_auc_score(y_val,gaussian_predictions)
auc
```

[10]:
```
0.821321195194423
```

[11]:
```
plt.figure(figsize=(10,5))
plt.plot(fpr,tpr,label='AUC:'+str(auc))
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('roc_curve of Naive Bayse (AUC=%.4f)' %(auc))
plt.legend(loc=4)
plt.grid()
```



[12]:
```
Y_pred_gaussian= gaussian.predict(test_df)
```

[13]:
```
submission_gaussian = pd.DataFrame({'activity_id' : test_id, 'outcome':␣
↪Y_pred_gaussian})
submission_gaussian.to_csv('submission_gaussian.csv', index = False)
```

[14]:
```
# kaggle score of Random Forest: 0.67487
```