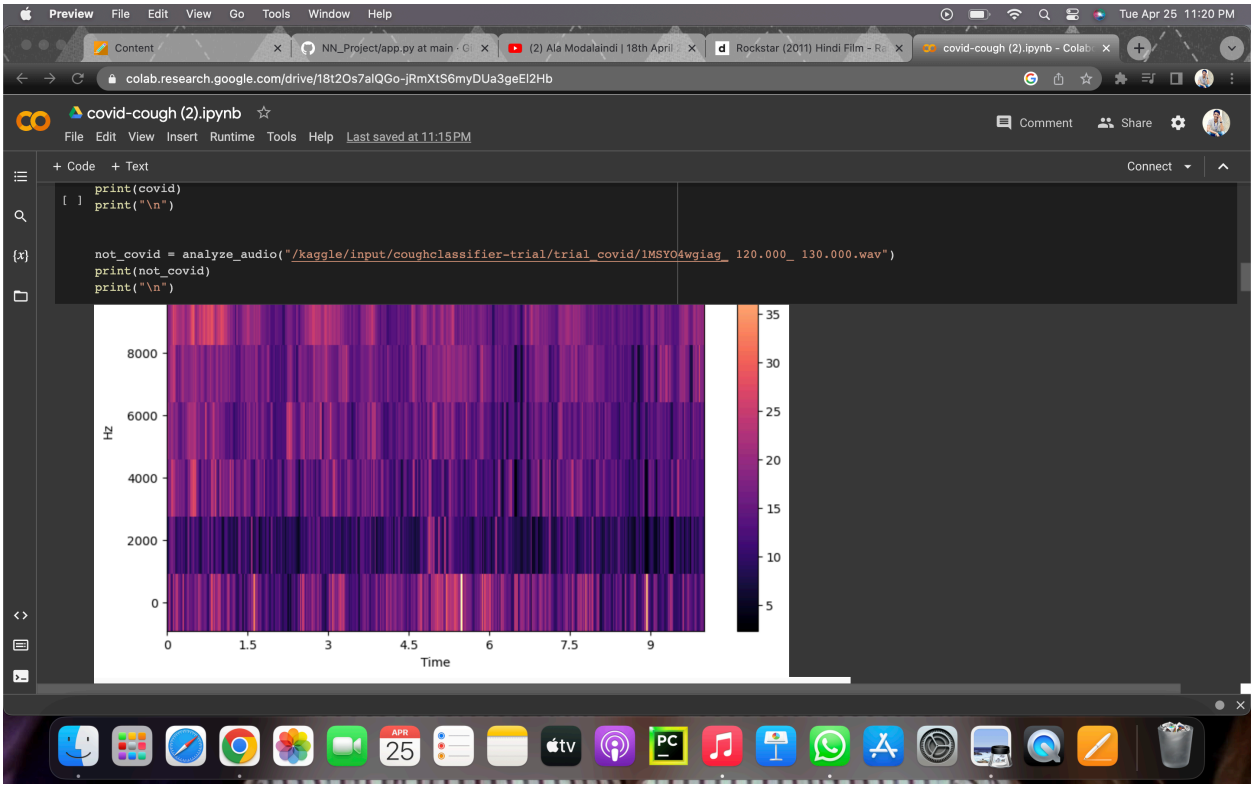
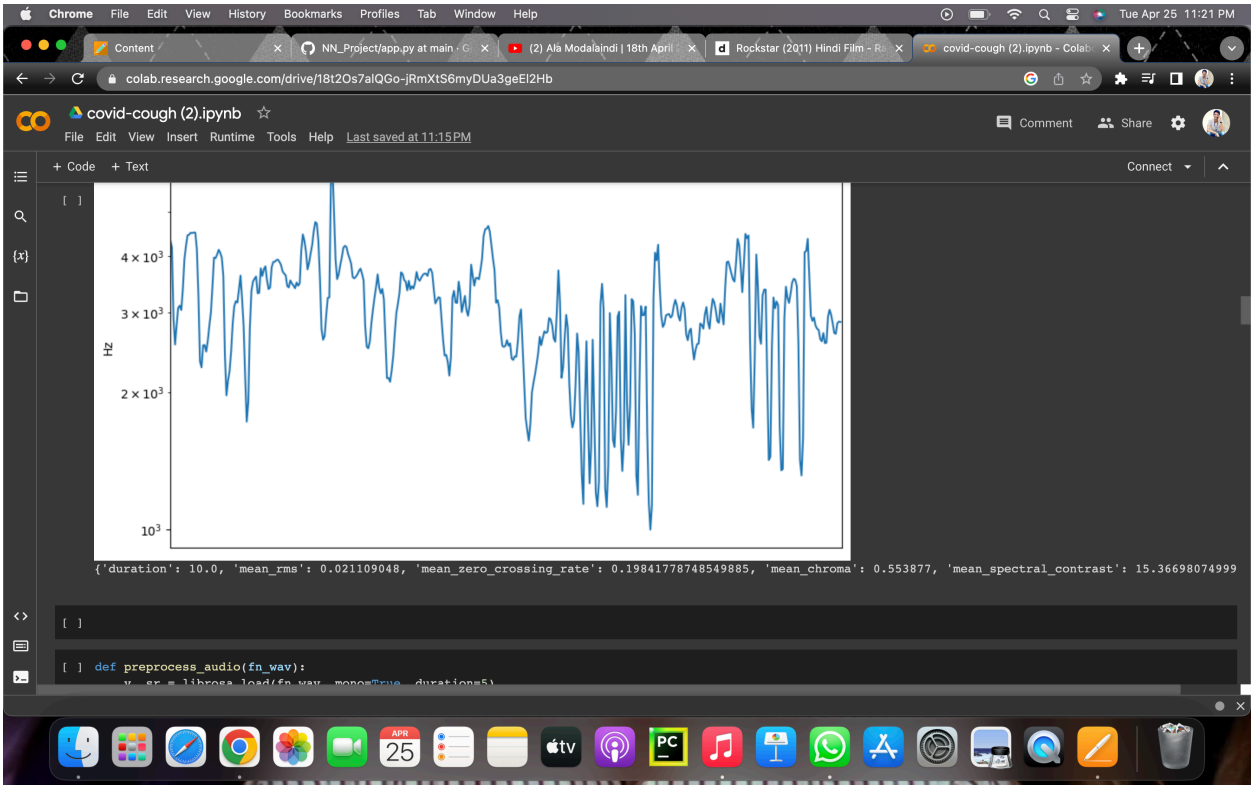
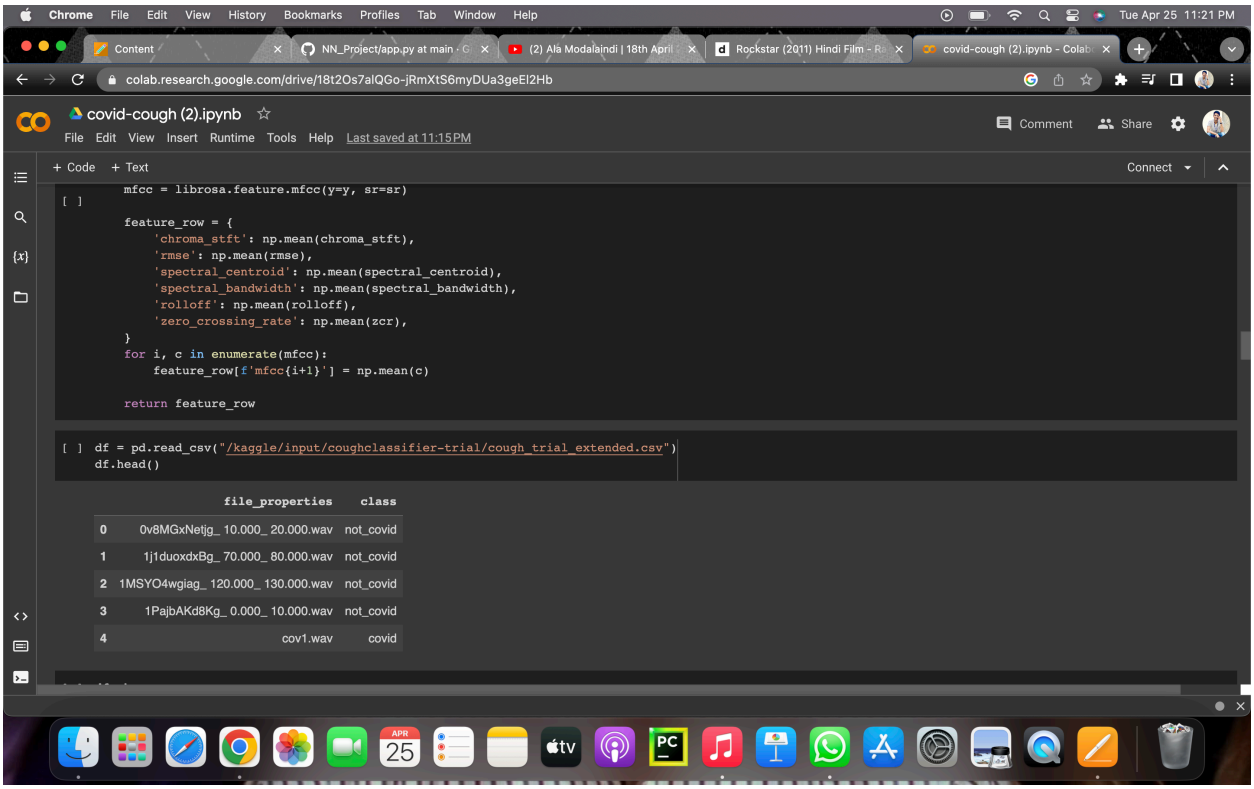
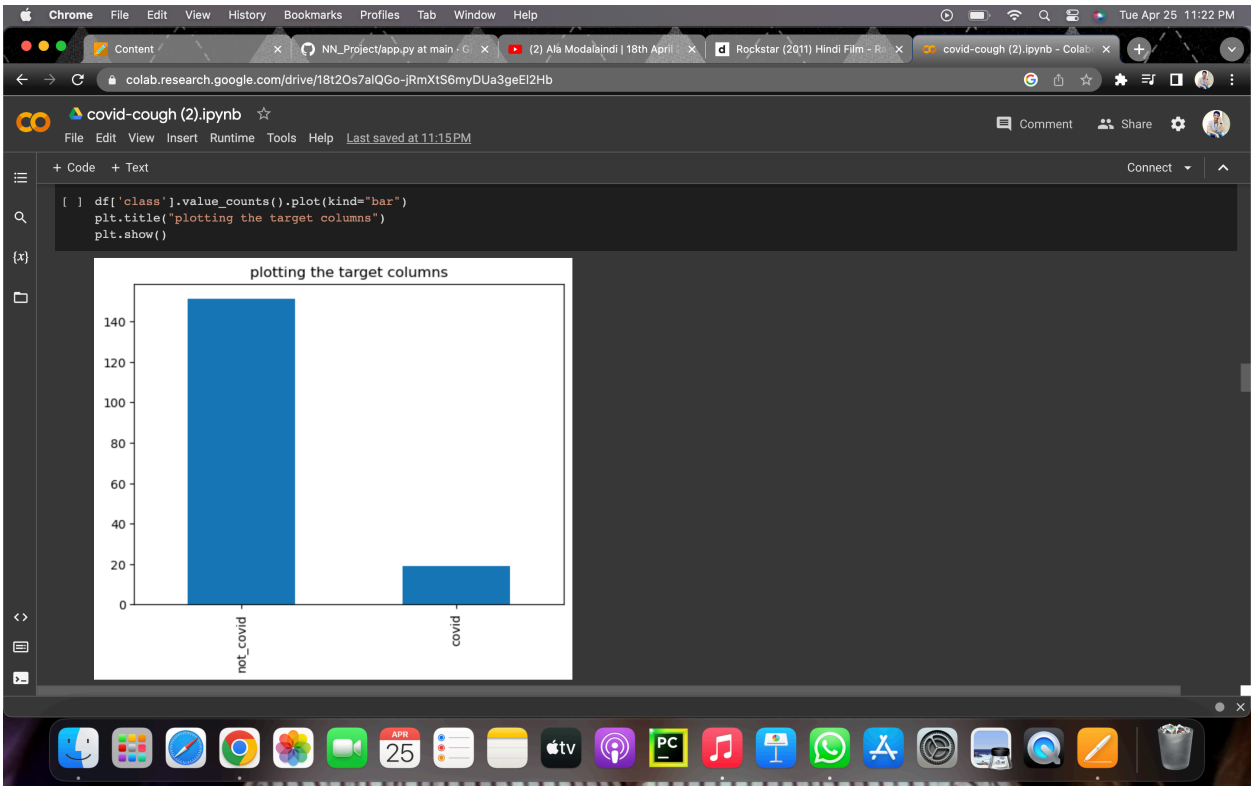
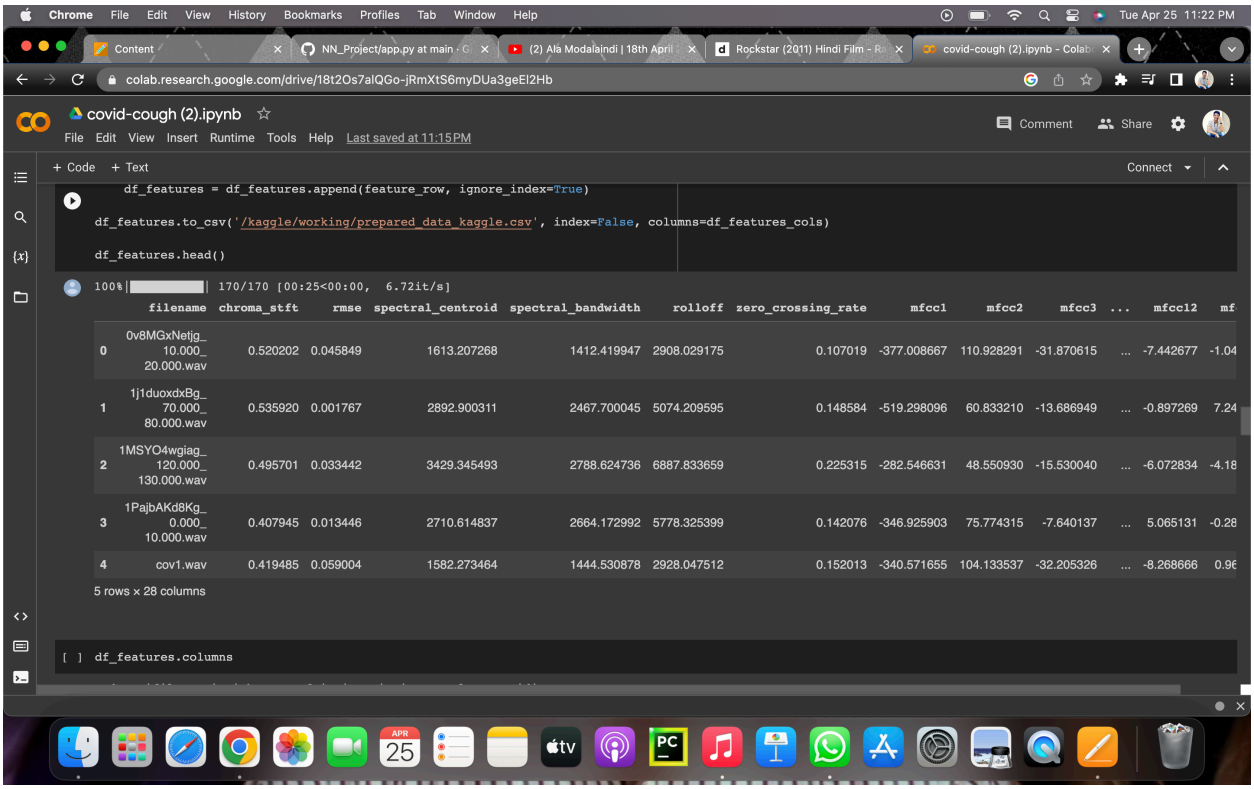
EXECUTION SCREENSHOTS






Chrome File Edit View History Bookmarks Profiles Tab Window Help
Tue Apr 25 11:21 PM
colab.research.google.com/drive/18t2Os7alQGo-jRmXtS6myDUa3geEl2Hb
covid-cough (2).ipynb
File Edit View Insert Runtime Tools Help Last saved at 11:15 PM
Comment
Share
Connect
+ Code + Text
mfcc = librosa.feature.mfcc(y=y, sr=sr)
feature_row = {
'chroma_stft': np.mean(chroma_stft),
'rmse': np.mean(rmse),
'spectral_centroid': np.mean(spectral_centroid),
'spectral_bandwidth': np.mean(spectral_bandwidth),
'rolloff': np.mean(rolloff),
'zero_crossing_rate': np.mean(zcr),
}
for i, c in enumerate(mfcc):
feature_row[f'mfcc{i+1}'] = np.mean(c)
return feature_row
df = pd.read_csv("/kaggle/input/coughclassifier-trial/cough_trial_extended.csv")
df.head()
file_properties class
0 0v8MGxNetjg_ 10.000_ 20.000.wav not_covid
1 1j1duoxdxBg_ 70.000_ 80.000.wav not_covid
2 1MSYO4wgiag_ 120.000_ 130.000.wav not_covid
3 1PajbAKd8Kg_ 0.000_ 10.000.wav not_covid
4 cov1.wav covid


Chrome File Edit View History Bookmarks Profiles Tab Window Help
Tue Apr 25 11:22 PM
colab.research.google.com/drive/18t2Os7alQGo-jRmXtS6myDUa3geEl2Hb
covid-cough (2).ipynb
File Edit View Insert Runtime Tools Help Last saved at 11:15 PM
Comment
Share
Connect
+ Code + Text
df['class'].value_counts().plot(kind="bar")
plt.title("plotting the target columns")
plt.show()
plotting the target columns
140
120
100
80
60
40
20
0
not_covid
covid



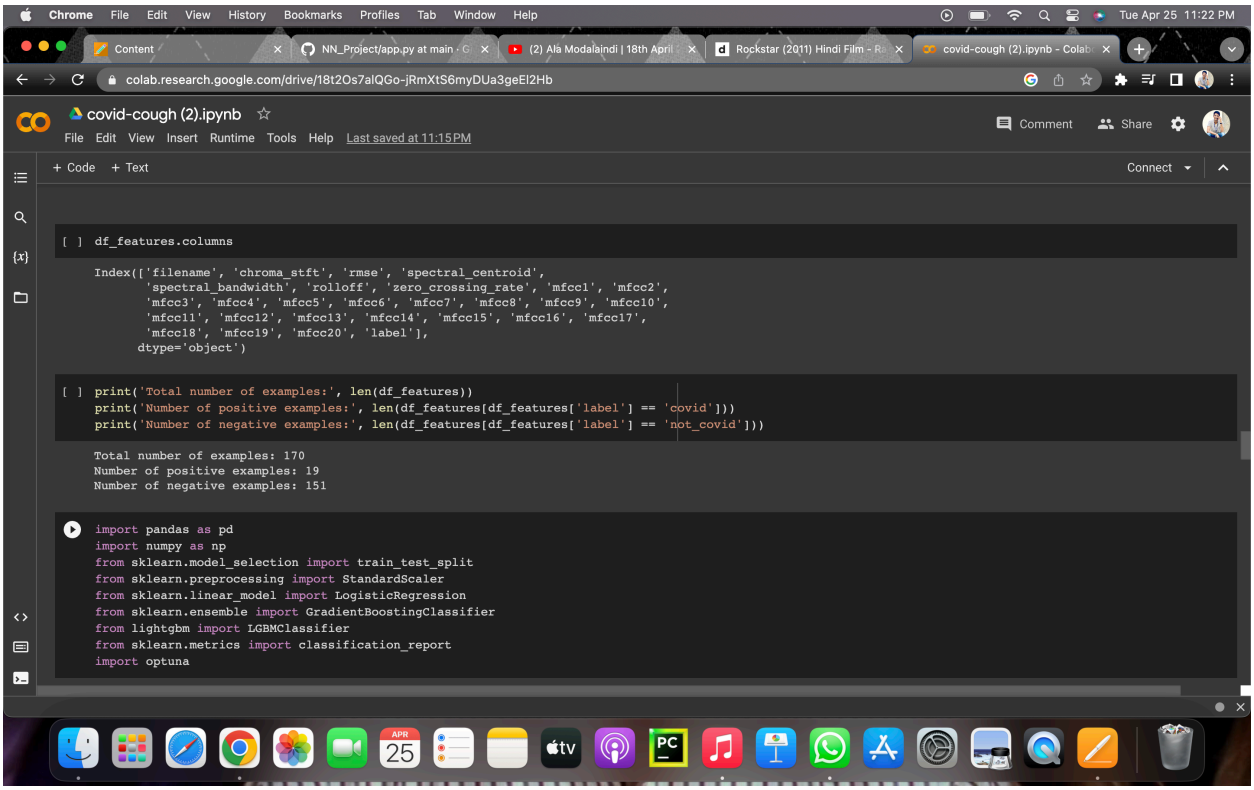
```
    df_features = df_features.append(feature_row, ignore_index=True)

df_features.to_csv('/kaggle/working/prepared_data_kaggle.csv', index=False, columns=df_features_cols)

df_features.head()
```

```
100%|██████████| 170/170 [00:25<00:00,  6.72it/s]
```

	filename	chroma_stft	rmse	spectral_centroid	spectral_bandwidth	rolloff	zero_crossing_rate	mfcc1	mfcc2	mfcc3	...	mfcc12	mf
0	0v8MGxNetjg_10.000_20.000.wav	0.520202	0.045849	1613.207268	1412.419947	2908.029175	0.107019	-377.008667	110.928291	-31.870615	...	-7.442677	-1.04
1	1j1duoxdxBg_70.000_80.000.wav	0.535920	0.001767	2892.900311	2467.700045	5074.209595	0.148584	-519.298096	60.833210	-13.686949	...	-0.897269	7.24
2	1MSYO4wgiag_120.000_130.000.wav	0.495701	0.033442	3429.345493	2788.624736	6887.833659	0.225315	-282.546631	48.550930	-15.530040	...	-6.072834	-4.18
3	1PajbAKd8Kg_0.000_10.000.wav	0.407945	0.013446	2710.614837	2664.172992	5778.325399	0.142076	-346.925903	75.774315	-7.640137	...	5.065131	-0.28
4	cov1.wav	0.419485	0.059004	1582.273464	1444.530878	2928.047512	0.152013	-340.571655	104.133537	-32.205326	...	-8.268666	0.96

5 rows x 28 columns

```
[ ] df_features.columns
```



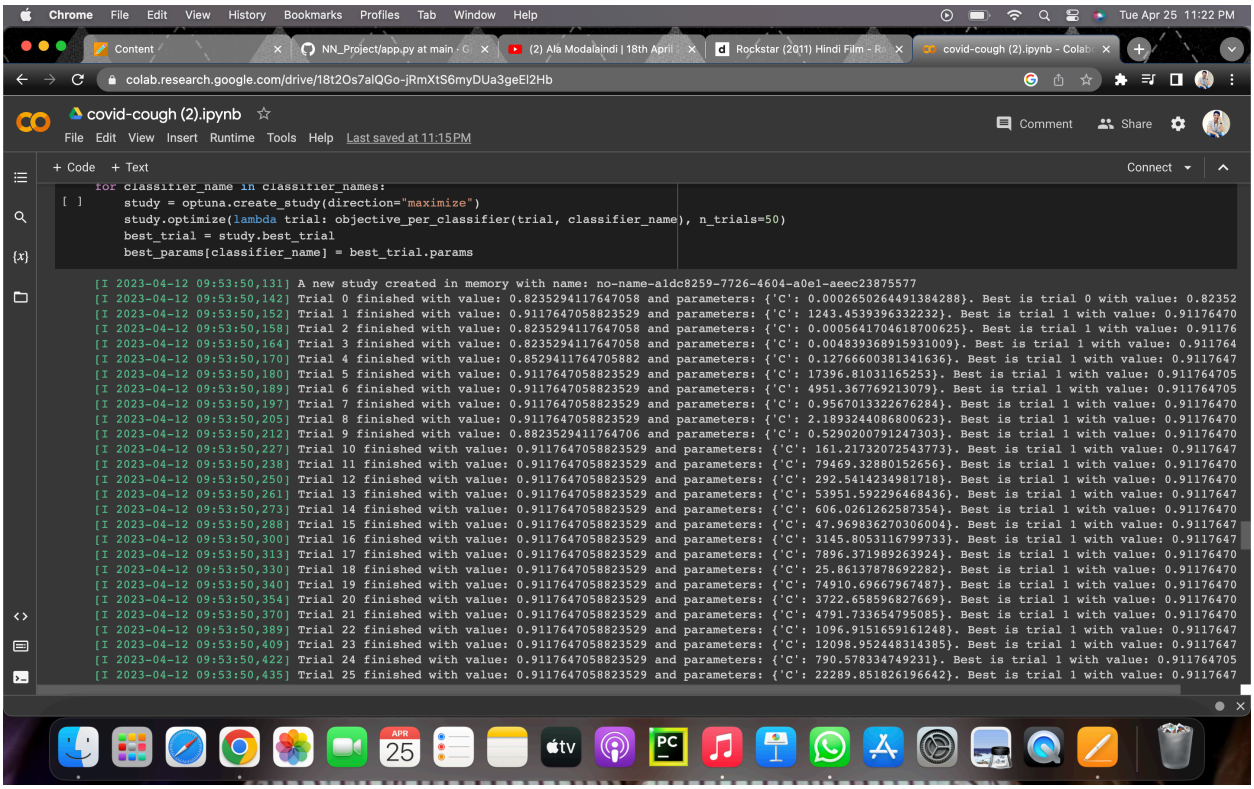
```
[ ] df_features.columns
```

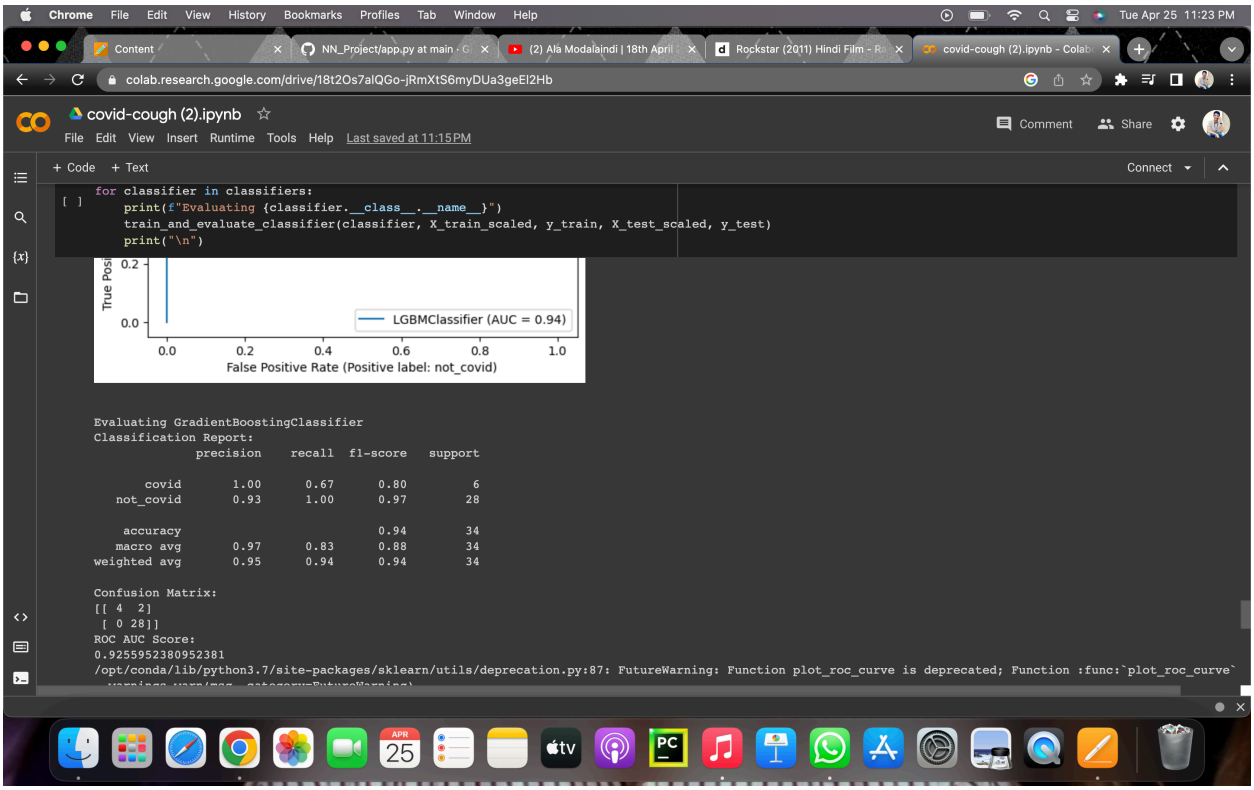
```
Index(['filename', 'chroma_stft', 'rmse', 'spectral_centroid',
       'spectral_bandwidth', 'rolloff', 'zero_crossing_rate', 'mfcc1', 'mfcc2',
       'mfcc3', 'mfcc4', 'mfcc5', 'mfcc6', 'mfcc7', 'mfcc8', 'mfcc9', 'mfcc10',
       'mfcc11', 'mfcc12', 'mfcc13', 'mfcc14', 'mfcc15', 'mfcc16', 'mfcc17',
       'mfcc18', 'mfcc19', 'mfcc20', 'label'],
      dtype='object')
```

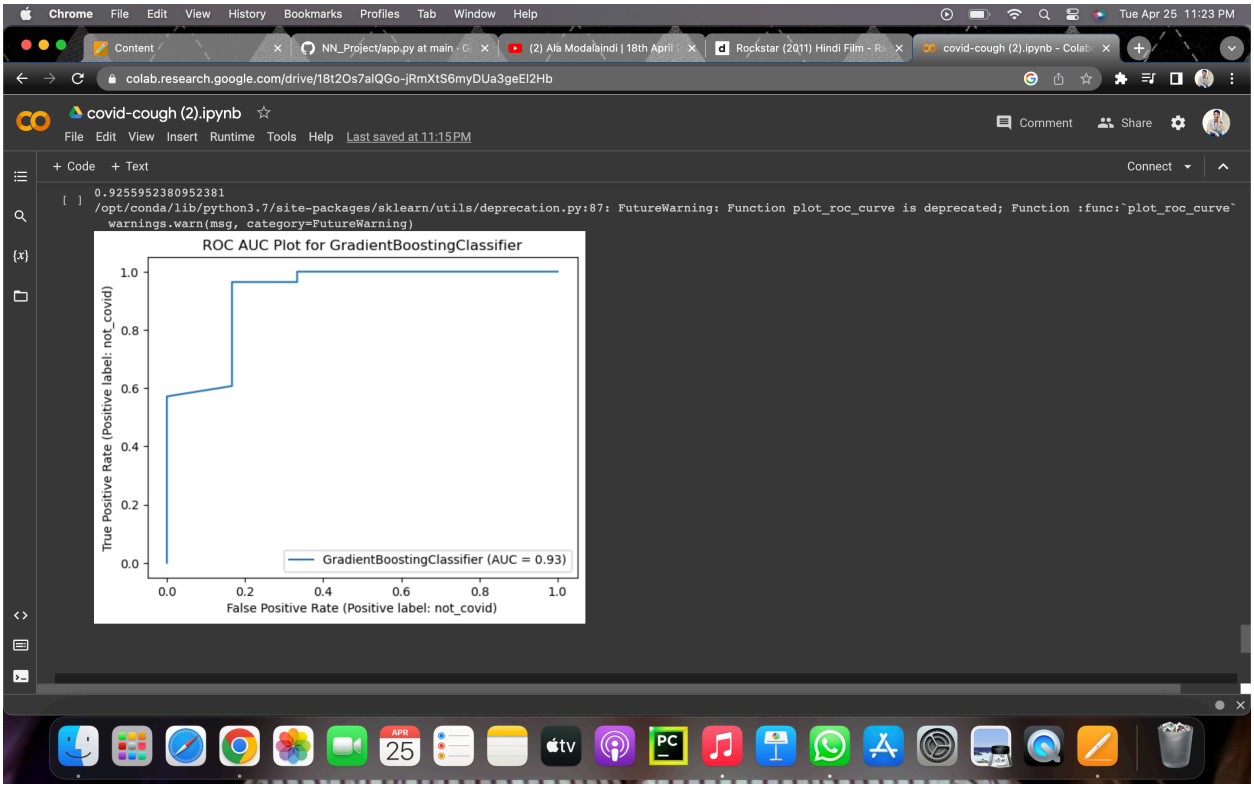
```
[ ] print('Total number of examples:', len(df_features))
    print('Number of positive examples:', len(df_features[df_features['label'] == 'covid']))
    print('Number of negative examples:', len(df_features[df_features['label'] == 'not_covid']))
```

```
Total number of examples: 170
Number of positive examples: 19
Number of negative examples: 151
```

```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import GradientBoostingClassifier
from lightgbm import LGBMClassifier
from sklearn.metrics import classification_report
import optuna
```


Chrome File Edit View History Bookmarks Profiles Tab Window Help
Tue Apr 25 11:22 PM
colab.research.google.com/drive/18t2Os7alQGo-jRmXtS6myDUa3geEl2Hb
covid-cough (2).ipynb
File Edit View Insert Runtime Tools Help Last saved at 11:15 PM
Comment Share
+ Code + Text
Connect
for classifier_name in classifier_names:
study = optuna.create_study(direction="maximize")
study.optimize(lambda trial: objective_per_classifier(trial, classifier_name), n_trials=50)
best_trial = study.best_trial
best_params[classifier_name] = best_trial.params
[I 2023-04-12 09:53:50,131] A new study created in memory with name: no-name-a1dc8259-7726-4604-a0e1-aeec23875577
[I 2023-04-12 09:53:50,142] Trial 0 finished with value: 0.8235294117647058 and parameters: {'C': 0.0002650264491384288}. Best is trial 0 with value: 0.82352
[I 2023-04-12 09:53:50,152] Trial 1 finished with value: 0.9117647058823529 and parameters: {'C': 1243.4539396332232}. Best is trial 1 with value: 0.91176470
[I 2023-04-12 09:53:50,158] Trial 2 finished with value: 0.8235294117647058 and parameters: {'C': 0.0005641704618700625}. Best is trial 1 with value: 0.91176
[I 2023-04-12 09:53:50,164] Trial 3 finished with value: 0.8235294117647058 and parameters: {'C': 0.004839368915931009}. Best is trial 1 with value: 0.911764
[I 2023-04-12 09:53:50,170] Trial 4 finished with value: 0.8529411764705882 and parameters: {'C': 0.12766600381341636}. Best is trial 1 with value: 0.9117647
[I 2023-04-12 09:53:50,180] Trial 5 finished with value: 0.9117647058823529 and parameters: {'C': 17396.81031165253}. Best is trial 1 with value: 0.911764705
[I 2023-04-12 09:53:50,189] Trial 6 finished with value: 0.9117647058823529 and parameters: {'C': 4951.367769213079}. Best is trial 1 with value: 0.911764705
[I 2023-04-12 09:53:50,197] Trial 7 finished with value: 0.9117647058823529 and parameters: {'C': 0.9567013322676284}. Best is trial 1 with value: 0.91176470
[I 2023-04-12 09:53:50,205] Trial 8 finished with value: 0.9117647058823529 and parameters: {'C': 2.1893244086800623}. Best is trial 1 with value: 0.91176470
[I 2023-04-12 09:53:50,212] Trial 9 finished with value: 0.8823529411764706 and parameters: {'C': 0.5290200791247303}. Best is trial 1 with value: 0.91176470
[I 2023-04-12 09:53:50,227] Trial 10 finished with value: 0.9117647058823529 and parameters: {'C': 161.21732072543773}. Best is trial 1 with value: 0.9117647
[I 2023-04-12 09:53:50,238] Trial 11 finished with value: 0.9117647058823529 and parameters: {'C': 79469.32880152656}. Best is trial 1 with value: 0.91176470
[I 2023-04-12 09:53:50,250] Trial 12 finished with value: 0.9117647058823529 and parameters: {'C': 292.5414234981718}. Best is trial 1 with value: 0.91176470
[I 2023-04-12 09:53:50,261] Trial 13 finished with value: 0.9117647058823529 and parameters: {'C': 53951.592296468436}. Best is trial 1 with value: 0.9117647
[I 2023-04-12 09:53:50,273] Trial 14 finished with value: 0.9117647058823529 and parameters: {'C': 606.0261262587354}. Best is trial 1 with value: 0.91176470
[I 2023-04-12 09:53:50,288] Trial 15 finished with value: 0.9117647058823529 and parameters: {'C': 47.969836270306004}. Best is trial 1 with value: 0.9117647
[I 2023-04-12 09:53:50,300] Trial 16 finished with value: 0.9117647058823529 and parameters: {'C': 3145.8053116799733}. Best is trial 1 with value: 0.9117647
[I 2023-04-12 09:53:50,313] Trial 17 finished with value: 0.9117647058823529 and parameters: {'C': 7896.371989263924}. Best is trial 1 with value: 0.91176470
[I 2023-04-12 09:53:50,330] Trial 18 finished with value: 0.9117647058823529 and parameters: {'C': 25.86137878692282}. Best is trial 1 with value: 0.91176470
[I 2023-04-12 09:53:50,340] Trial 19 finished with value: 0.9117647058823529 and parameters: {'C': 74910.69667967487}. Best is trial 1 with value: 0.91176470
[I 2023-04-12 09:53:50,354] Trial 20 finished with value: 0.9117647058823529 and parameters: {'C': 3722.658596827669}. Best is trial 1 with value: 0.91176470
[I 2023-04-12 09:53:50,370] Trial 21 finished with value: 0.9117647058823529 and parameters: {'C': 4791.733654795085}. Best is trial 1 with value: 0.91176470
[I 2023-04-12 09:53:50,389] Trial 22 finished with value: 0.9117647058823529 and parameters: {'C': 1096.9151659161248}. Best is trial 1 with value: 0.9117647
[I 2023-04-12 09:53:50,409] Trial 23 finished with value: 0.9117647058823529 and parameters: {'C': 12098.952448314385}. Best is trial 1 with value: 0.9117647
[I 2023-04-12 09:53:50,422] Trial 24 finished with value: 0.9117647058823529 and parameters: {'C': 790.578334749231}. Best is trial 1 with value: 0.9117647
[I 2023-04-12 09:53:50,435] Trial 25 finished with value: 0.9117647058823529 and parameters: {'C': 22289.851826196642}. Best is trial 1 with value: 0.9117647


Chrome File Edit View History Bookmarks Profiles Tab Window Help
Tue Apr 25 11:23 PM
colab.research.google.com/drive/18t2Os7alQGo-jRmXtS6myDUa3geEl2Hb
covid-cough (2).ipynb
File Edit View Insert Runtime Tools Help Last saved at 11:15 PM
Comment Share
+ Code + Text
Connect
for classifier in classifiers:
print(f"Evaluating {classifier.__class__.__name__}")
train_and_evaluate_classifier(classifier, X_train_scaled, y_train, X_test_scaled, y_test)
print("\n")
True Positive Rate
0.2
0.0
LGBMClassifier (AUC = 0.94)
0.0
0.2
0.4
0.6
0.8
1.0
False Positive Rate (Positive label: not_covid)
Evaluating GradientBoostingClassifier
Classification Report:
precision recall f1-score support
covid 1.00 0.67 0.80 6
not_covid 0.93 1.00 0.97 28
accuracy 0.94 34
macro avg 0.97 0.83 0.88 34
weighted avg 0.95 0.94 0.94 34
Confusion Matrix:
[[4 2]
[0 28]]
ROC AUC Score:
0.9255952380952381
/opt/conda/lib/python3.7/site-packages/sklearn/utils/deprecation.py:87: FutureWarning: Function plot_roc_curve is deprecated; Function :func:`plot_roc_curve`


Chrome File Edit View History Bookmarks Profiles Tab Window Help
Tue Apr 25 11:23 PM
Content
NN_Project/app.py at main · C
(2) Ala Modalaindi | 18th April
Rockstar (2011) Hindi Film - R
covid-cough (2).ipynb - Colab
colab.research.google.com/drive/18t2Os7alQGo-jRmXtS6myDUa3geEl2Hb
covid-cough (2).ipynb
File Edit View Insert Runtime Tools Help Last saved at 11:15 PM
Comment
Share
+ Code + Text
Connect
[ ] 0.9255952380952381
/opt/conda/lib/python3.7/site-packages/sklearn/utils/deprecation.py:87: FutureWarning: Function plot_roc_curve is deprecated; Function :func:`plot_roc_curve`
warnings.warn(msg, category=FutureWarning)
ROC AUC Plot for GradientBoostingClassifier
True Positive Rate (Positive label: not_covid)
False Positive Rate (Positive label: not_covid)
GradientBoostingClassifier (AUC = 0.93)
0.0
0.2
0.4
0.6
0.8
1.0