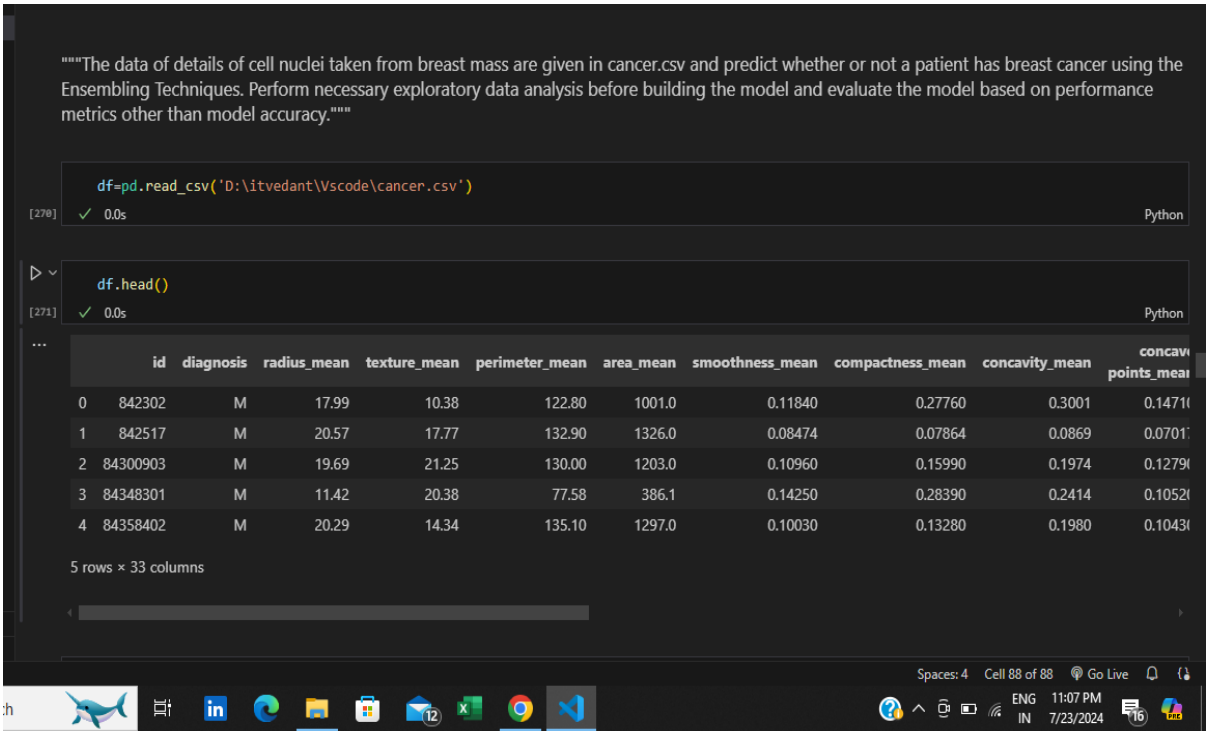
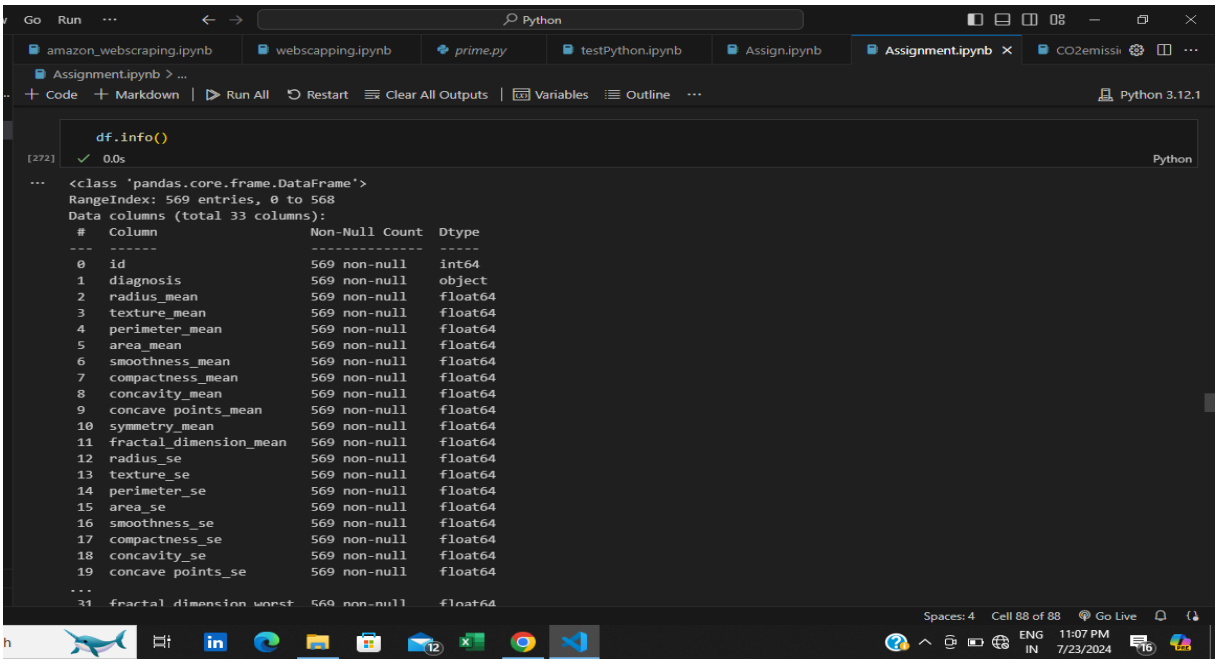
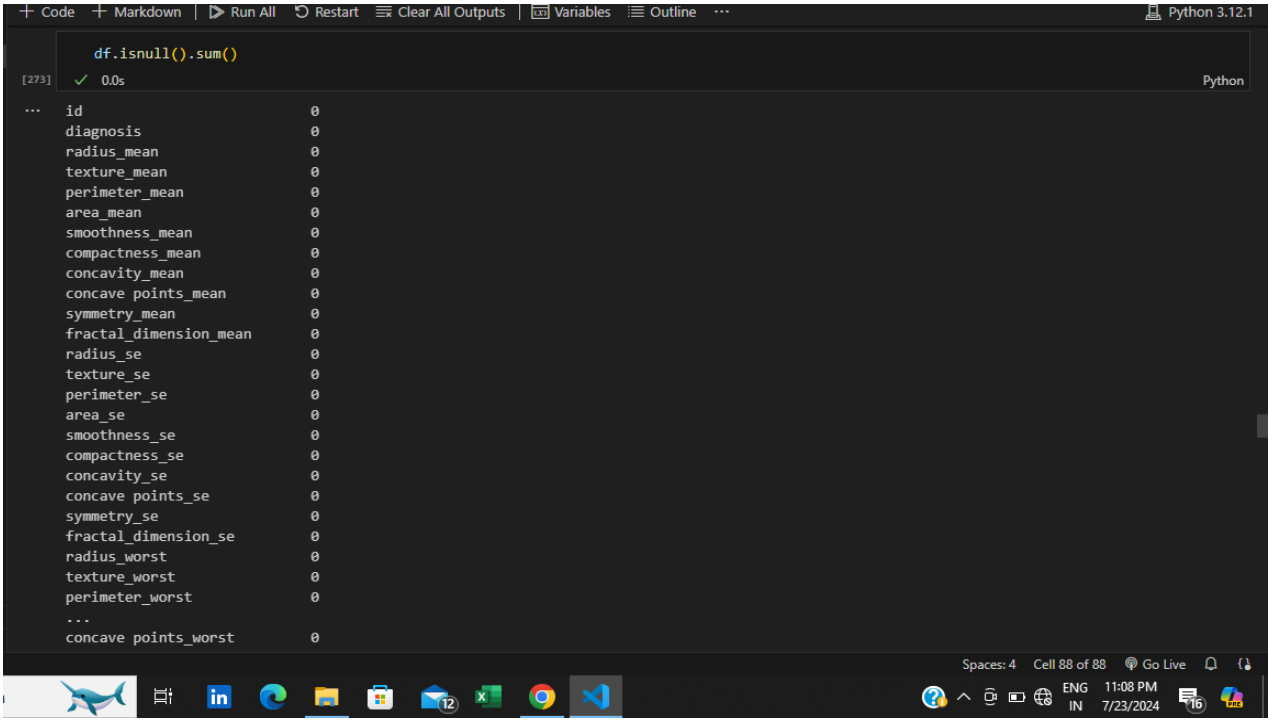
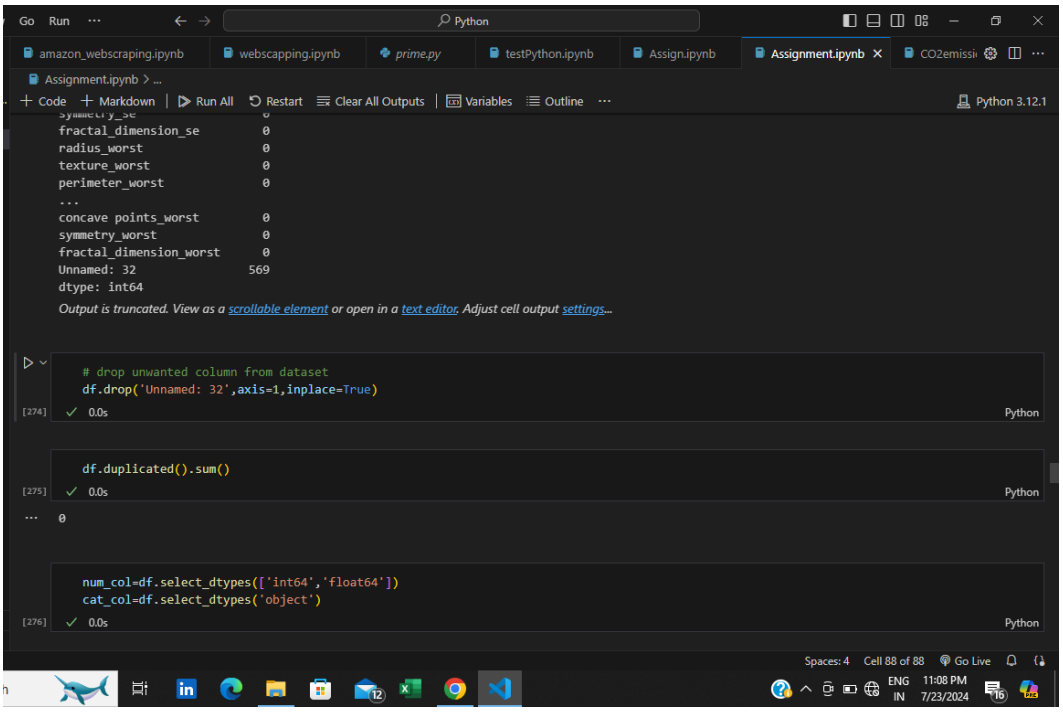
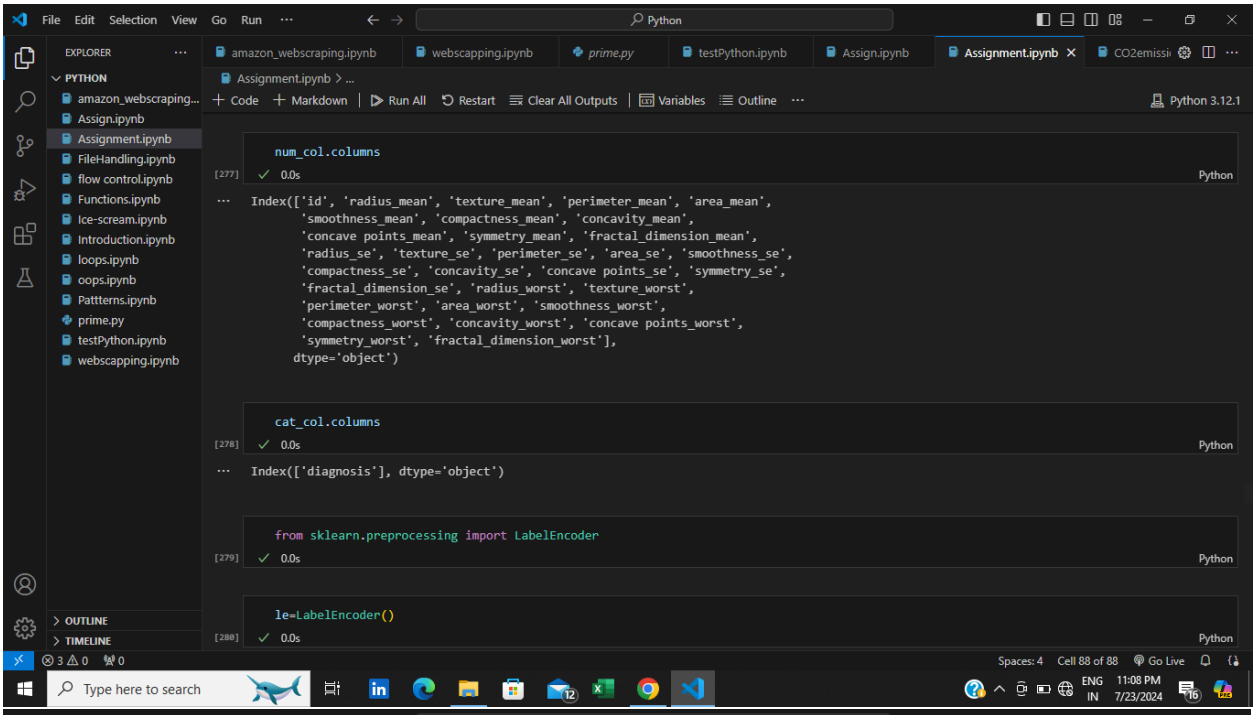
The data of details of cell nuclei taken from breast mass are given in cancer.csv and predict whether or not a patient has breast cancer using the Ensembling Techniques. Perform necessary exploratory data analysis before building the model and evaluate the model based on performance metrics other than model accuracy.

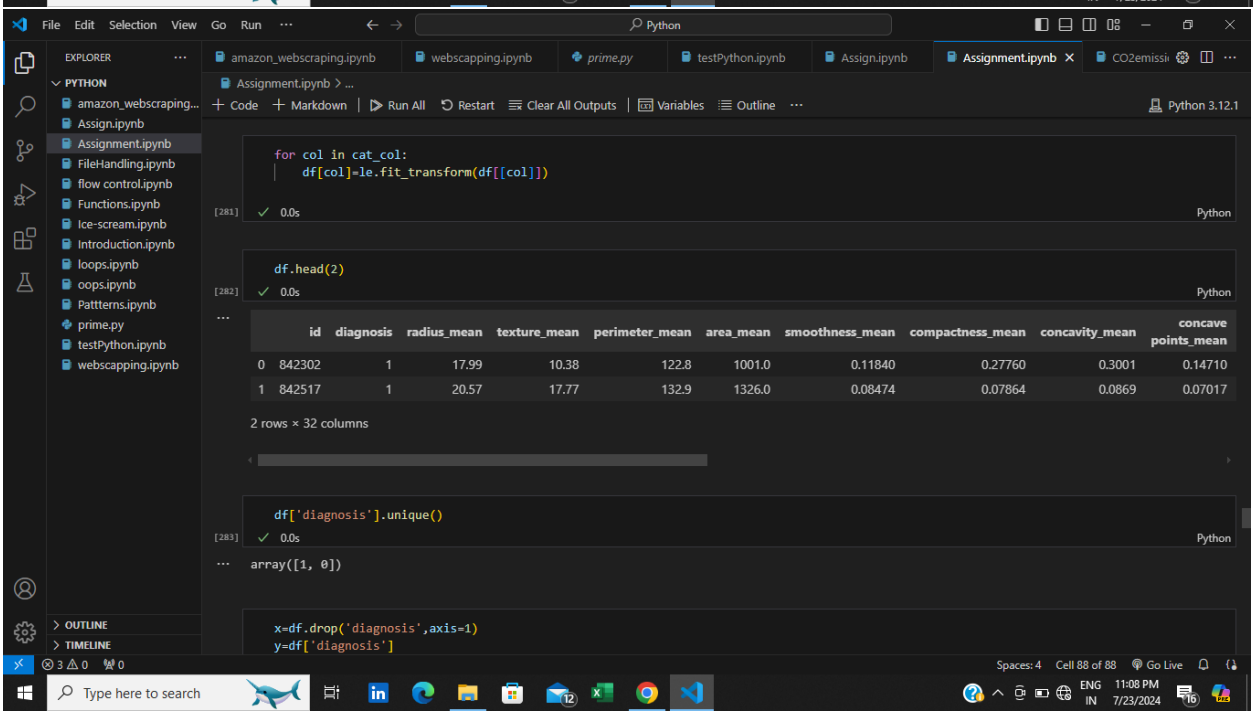


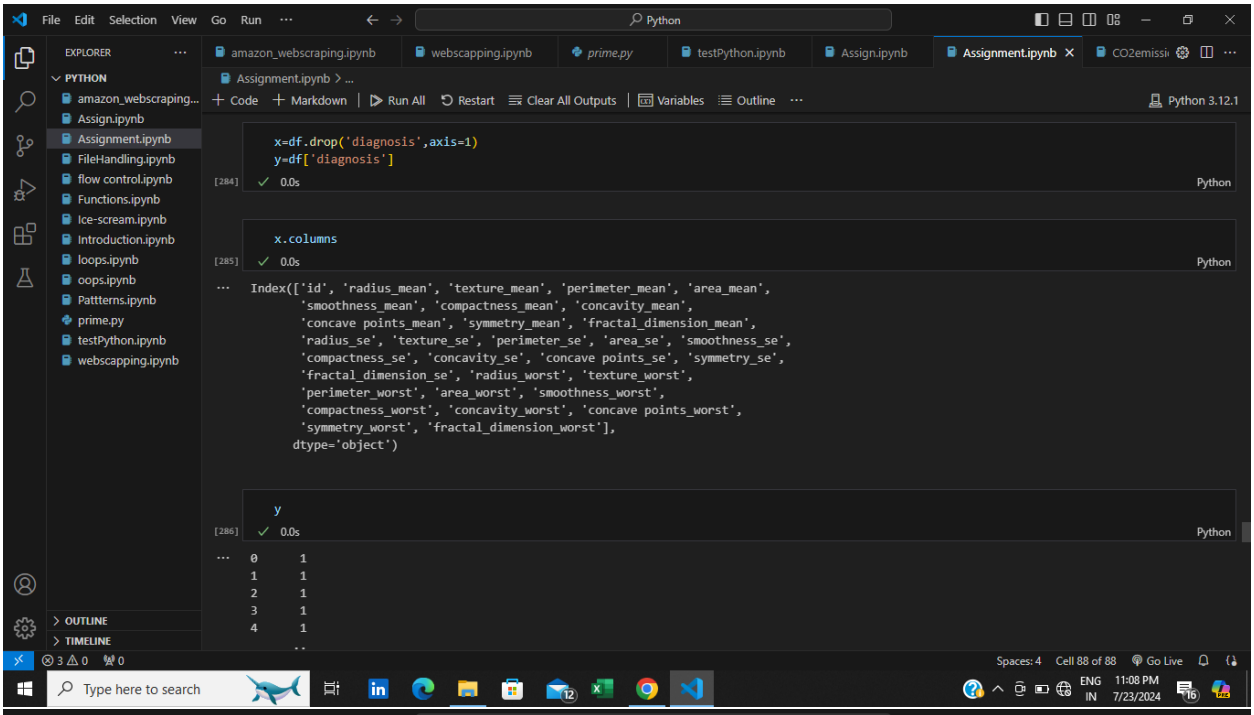


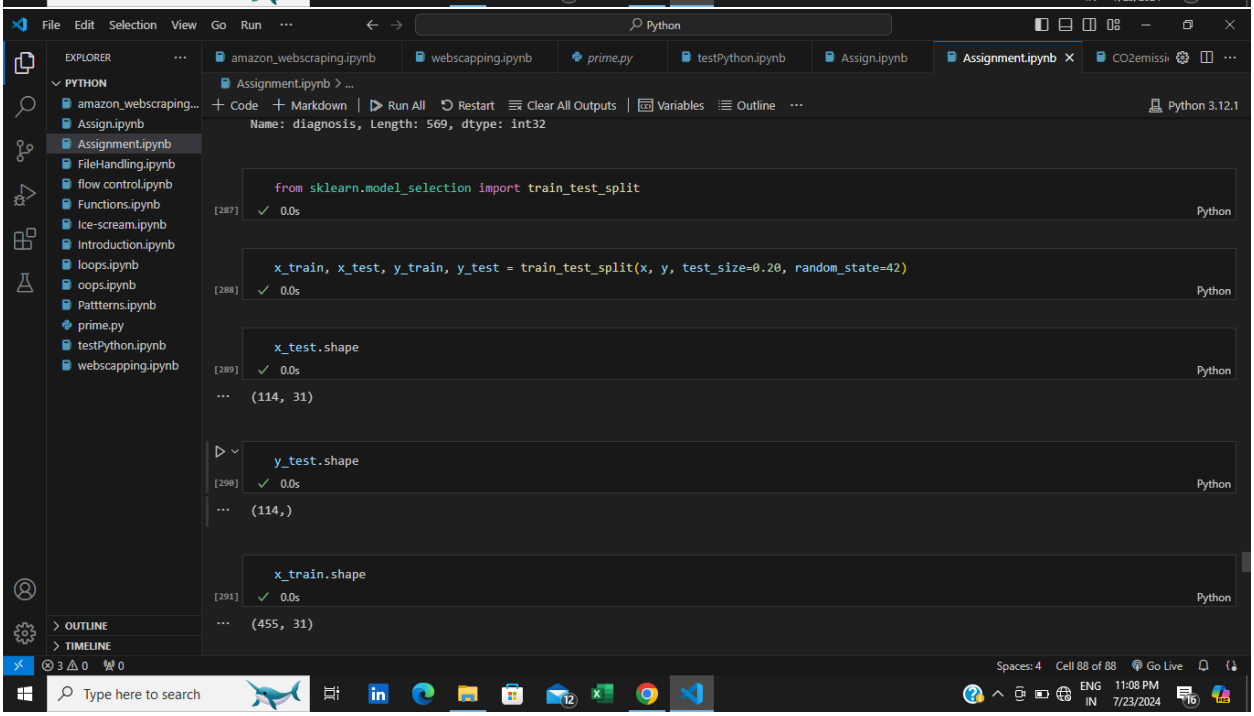

Code
Markdown
Run All
Restart
Clear All Outputs
Variables
Outline
Python 3.12.1
df.isnull().sum()
[273]
0.0s
Python
id 0
diagnosis 0
radius_mean 0
texture_mean 0
perimeter_mean 0
area_mean 0
smoothness_mean 0
compactness_mean 0
concavity_mean 0
concave points_mean 0
symmetry_mean 0
fractal_dimension_mean 0
radius_se 0
texture_se 0
perimeter_se 0
area_se 0
smoothness_se 0
compactness_se 0
concavity_se 0
concave points_se 0
symmetry_se 0
fractal_dimension_se 0
radius_worst 0
texture_worst 0
perimeter_worst 0
...
concave points_worst 0
Spaces: 4
Cell 88 of 88
Go Live
ENG
IN
11:08 PM
7/23/2024

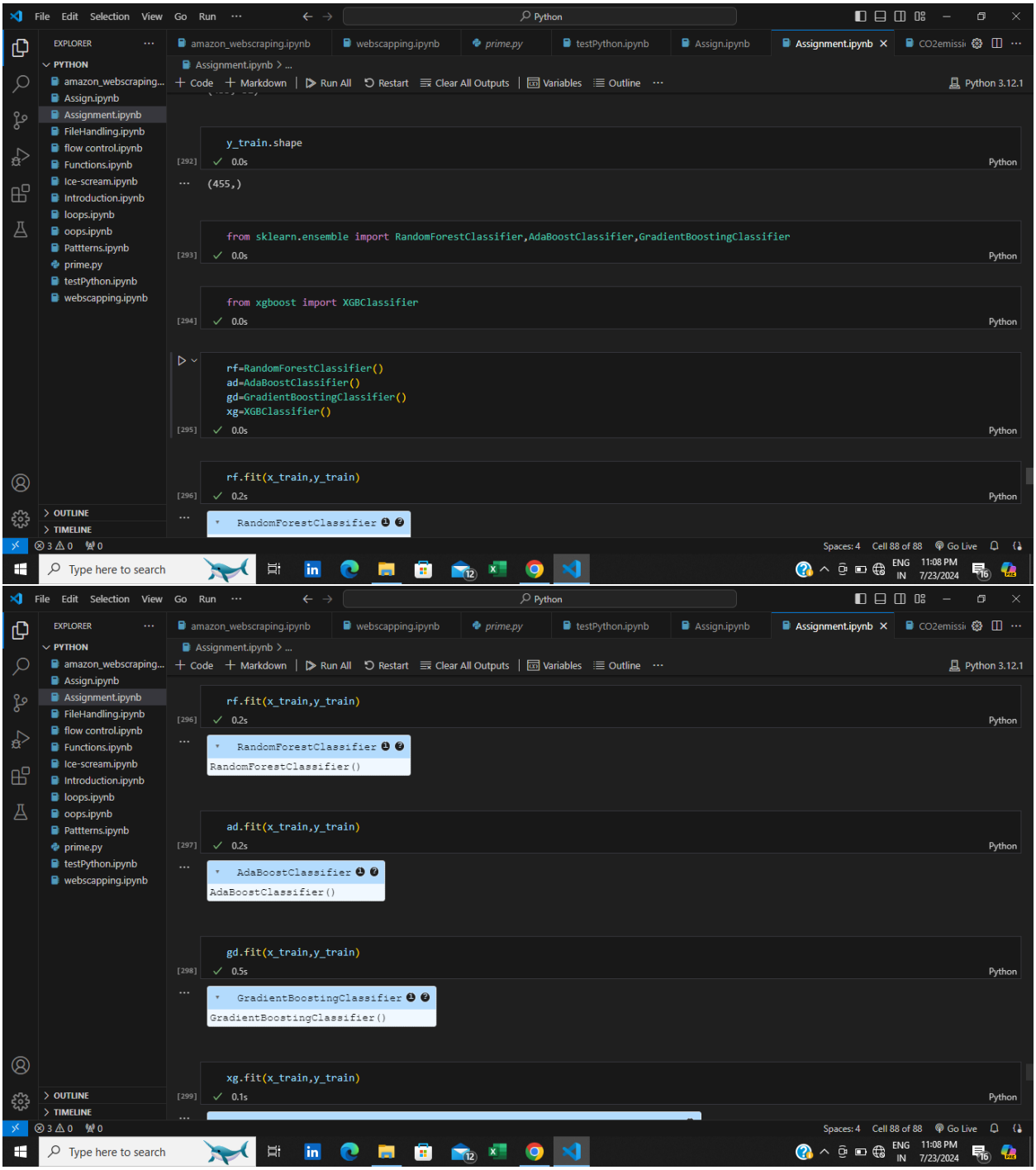

Go
Run
Python
amazon_webscraping.ipynb
webscapping.ipynb
prime.py
testPython.ipynb
Assign.ipynb
Assignment.ipynb
CO2emissi
Assignment.ipynb > ...
Code
Markdown
Run All
Restart
Clear All Outputs
Variables
Outline
Python 3.12.1
fractal_dimension_se 0
radius_worst 0
texture_worst 0
perimeter_worst 0
...
concave points_worst 0
symmetry_worst 0
fractal_dimension_worst 0
Unnamed: 32 569
dtype: int64
Output is truncated. View as a scrollable element or open in a text editor. Adjust cell output settings...
# drop unwanted column from dataset
df.drop('Unnamed: 32',axis=1,inplace=True)
[274]
0.0s
Python
df.duplicated().sum()
[275]
0.0s
Python
0
num_col=df.select_dtypes(['int64','float64'])
cat_col=df.select_dtypes('object')
[276]
0.0s
Python
Spaces: 4
Cell 88 of 88
Go Live
ENG
IN
11:08 PM
7/23/2024


File Edit Selection View Go Run
Python
EXPLORER
PYTHON
amazon_webscraping...
Assign.ipynb
Assignment.ipynb
FileHandling.ipynb
flow control.ipynb
Functions.ipynb
Ice-scream.ipynb
Introduction.ipynb
loops.ipynb
oops.ipynb
Pattterns.ipynb
prime.py
testPython.ipynb
webscapping.ipynb
amazon_webscraping.ipynb
webscapping.ipynb
prime.py
testPython.ipynb
Assign.ipynb
Assignment.ipynb
CO2emissi
Assignment.ipynb > ...
Code
Markdown
Run All
Restart
Clear All Outputs
Variables
Outline
Python 3.12.1
num_col.columns
[277] 0.0s Python
Index(['id', 'radius_mean', 'texture_mean', 'perimeter_mean', 'area_mean',
'smoothness_mean', 'compactness_mean', 'concavity_mean',
'concave points_mean', 'symmetry_mean', 'fractal_dimension_mean',
'radius_se', 'texture_se', 'perimeter_se', 'area_se', 'smoothness_se',
'compactness_se', 'concavity_se', 'concave points_se', 'symmetry_se',
'fractal_dimension_se', 'radius_worst', 'texture_worst',
'perimeter_worst', 'area_worst', 'smoothness_worst',
'compactness_worst', 'concavity_worst', 'concave points_worst',
'symmetry_worst', 'fractal_dimension_worst'],
dtype='object')
cat_col.columns
[278] 0.0s Python
Index(['diagnosis'], dtype='object')
from sklearn.preprocessing import LabelEncoder
[279] 0.0s Python
le=LabelEncoder()
[280] 0.0s Python
OUTLINE
TIMELINE
Spaces: 4 Cell 88 of 88 Go Live
Type here to search
ENG IN 11:08 PM 7/23/2024

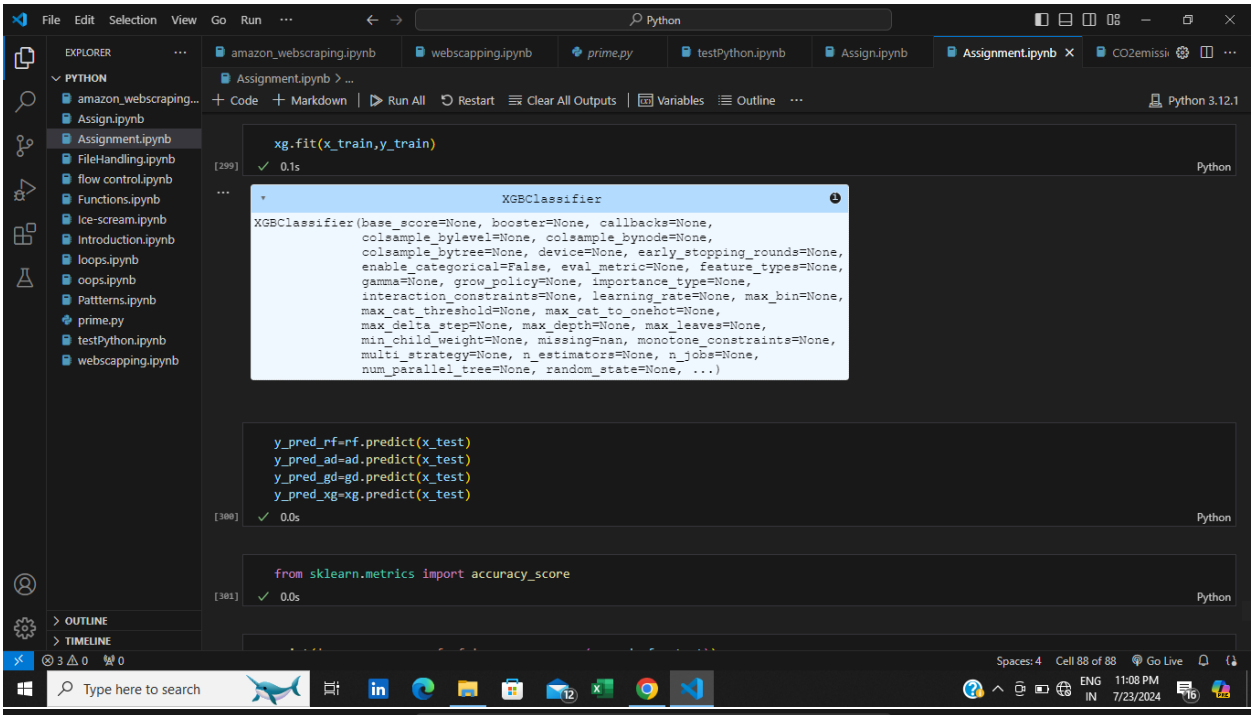

File Edit Selection View Go Run
Python
EXPLORER
PYTHON
amazon_webscraping...
Assign.ipynb
Assignment.ipynb
FileHandling.ipynb
flow control.ipynb
Functions.ipynb
Ice-scream.ipynb
Introduction.ipynb
loops.ipynb
oops.ipynb
Pattterns.ipynb
prime.py
testPython.ipynb
webscapping.ipynb
amazon_webscraping.ipynb
webscapping.ipynb
prime.py
testPython.ipynb
Assign.ipynb
Assignment.ipynb
CO2emissi
Assignment.ipynb > ...
Code
Markdown
Run All
Restart
Clear All Outputs
Variables
Outline
Python 3.12.1
for col in cat_col:
df[col]=le.fit_transform(df[[col]])
[281] 0.0s Python
df.head(2)
[282] 0.0s Python
id diagnosis radius_mean texture_mean perimeter_mean area_mean smoothness_mean compactness_mean concavity_mean concave points_mean
0 842302 1 17.99 10.38 122.8 1001.0 0.11840 0.27760 0.3001 0.14710
1 842517 1 20.57 17.77 132.9 1326.0 0.08474 0.07864 0.0869 0.07017
2 rows × 32 columns
df['diagnosis'].unique()
[283] 0.0s Python
array([1, 0])
x=df.drop('diagnosis',axis=1)
y=df['diagnosis']
OUTLINE
TIMELINE
Spaces: 4 Cell 88 of 88 Go Live
Type here to search
ENG IN 11:08 PM 7/23/2024

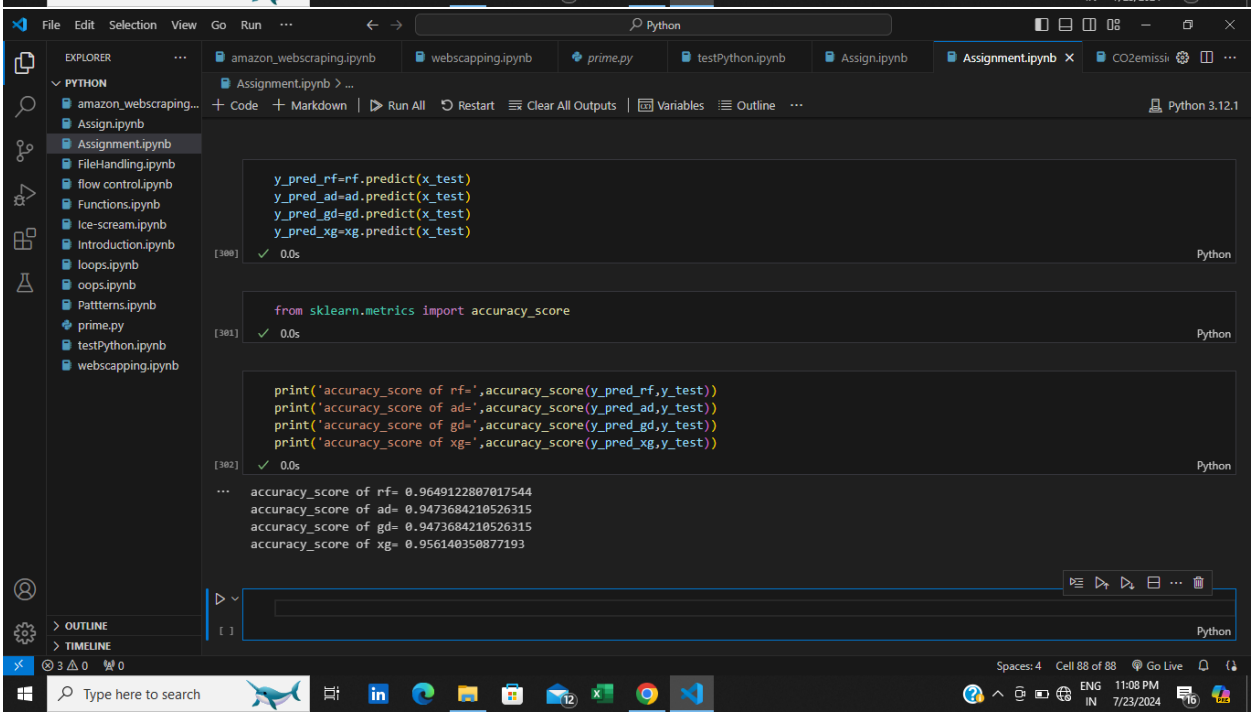

File Edit Selection View Go Run
Python
EXPLORER
PYTHON
amazon_webscraping...
Assign.ipynb
Assignment.ipynb
FileHandling.ipynb
flow control.ipynb
Functions.ipynb
Ice-scream.ipynb
Introduction.ipynb
loops.ipynb
oops.ipynb
Pattterns.ipynb
prime.py
testPython.ipynb
webscapping.ipynb
amazon_webscraping.ipynb
webscapping.ipynb
prime.py
testPython.ipynb
Assign.ipynb
Assignment.ipynb
CO2emissi
Assignment.ipynb > ...
Code Markdown Run All Restart Clear All Outputs Variables Outline
Python 3.12.1
x=df.drop('diagnosis',axis=1)
y=df['diagnosis']
[284] 0.0s Python
x.columns
[285] 0.0s Python
Index(['id', 'radius_mean', 'texture_mean', 'perimeter_mean', 'area_mean',
'smoothness_mean', 'compactness_mean', 'concavity_mean',
'concave points_mean', 'symmetry_mean', 'fractal_dimension_mean',
'radius_se', 'texture_se', 'perimeter_se', 'area_se', 'smoothness_se',
'compactness_se', 'concavity_se', 'concave points_se', 'symmetry_se',
'fractal_dimension_se', 'radius_worst', 'texture_worst',
'perimeter_worst', 'area_worst', 'smoothness_worst',
'compactness_worst', 'concavity_worst', 'concave points_worst',
'symmetry_worst', 'fractal_dimension_worst'],
dtype='object')
y
[286] 0.0s Python
0 1
1 1
2 1
3 1
4 1
OUTLINE
TIMELINE
Spaces: 4 Cell 88 of 88 Go Live
Type here to search
ENG IN 11:08 PM 7/23/2024


File Edit Selection View Go Run
Python
EXPLORER
PYTHON
amazon_webscraping...
Assign.ipynb
Assignment.ipynb
FileHandling.ipynb
flow control.ipynb
Functions.ipynb
Ice-scream.ipynb
Introduction.ipynb
loops.ipynb
oops.ipynb
Pattterns.ipynb
prime.py
testPython.ipynb
webscapping.ipynb
amazon_webscraping.ipynb
webscapping.ipynb
prime.py
testPython.ipynb
Assign.ipynb
Assignment.ipynb
CO2emissi
Assignment.ipynb > ...
Code Markdown Run All Restart Clear All Outputs Variables Outline
Python 3.12.1
Name: diagnosis, Length: 569, dtype: int32
from sklearn.model_selection import train_test_split
[287] 0.0s Python
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.20, random_state=42)
[288] 0.0s Python
x_test.shape
[289] 0.0s Python
(114, 31)
y_test.shape
[290] 0.0s Python
(114,)
x_train.shape
[291] 0.0s Python
(455, 31)
OUTLINE
TIMELINE
Spaces: 4 Cell 88 of 88 Go Live
Type here to search
ENG IN 11:08 PM 7/23/2024


File Edit Selection View Go Run
Python
EXPLORER
PYTHON
amazon_webscraping...
Assign.ipynb
Assignment.ipynb
FileHandling.ipynb
flow control.ipynb
Functions.ipynb
Ice-scream.ipynb
Introduction.ipynb
loops.ipynb
oops.ipynb
Pattterns.ipynb
prime.py
testPython.ipynb
webscapping.ipynb
amazon_webscraping.ipynb
webscapping.ipynb
prime.py
testPython.ipynb
Assign.ipynb
Assignment.ipynb
CO2emissi
Assignment.ipynb > ...
Code
Markdown
Run All
Restart
Clear All Outputs
Variables
Outline
Python 3.12.1
y_train.shape
[292] 0.0s Python
(455,)
from sklearn.ensemble import RandomForestClassifier,AdaBoostClassifier,GradientBoostingClassifier
[293] 0.0s Python
from xgboost import XGBClassifier
[294] 0.0s Python
rf=RandomForestClassifier()
ad=AdaBoostClassifier()
gd=GradientBoostingClassifier()
xg=XGBClassifier()
[295] 0.0s Python
rf.fit(x_train,y_train)
[296] 0.2s Python
RandomForestClassifier
OUTLINE
TIMELINE
Spaces: 4 Cell 88 of 88 Go Live
Type here to search
ENG IN 11:08 PM 7/23/2024

File Edit Selection View Go Run
Python
EXPLORER
PYTHON
amazon_webscraping...
Assign.ipynb
Assignment.ipynb
FileHandling.ipynb
flow control.ipynb
Functions.ipynb
Ice-scream.ipynb
Introduction.ipynb
loops.ipynb
oops.ipynb
Pattterns.ipynb
prime.py
testPython.ipynb
webscapping.ipynb
amazon_webscraping.ipynb
webscapping.ipynb
prime.py
testPython.ipynb
Assign.ipynb
Assignment.ipynb
CO2emissi
Assignment.ipynb > ...
Code
Markdown
Run All
Restart
Clear All Outputs
Variables
Outline
Python 3.12.1
rf.fit(x_train,y_train)
[296] 0.2s Python
RandomForestClassifier
RandomForestClassifier()
ad.fit(x_train,y_train)
[297] 0.2s Python
AdaBoostClassifier
AdaBoostClassifier()
gd.fit(x_train,y_train)
[298] 0.5s Python
GradientBoostingClassifier
GradientBoostingClassifier()
xg.fit(x_train,y_train)
[299] 0.1s Python
OUTLINE
TIMELINE
Spaces: 4 Cell 88 of 88 Go Live
Type here to search
ENG IN 11:08 PM 7/23/2024


File Edit Selection View Go Run
Python
EXPLORER
PYTHON
amazon_webscraping...
Assign.ipynb
Assignment.ipynb
FileHandling.ipynb
flow control.ipynb
Functions.ipynb
Ice-scream.ipynb
Introduction.ipynb
loops.ipynb
oops.ipynb
Pattterns.ipynb
prime.py
testPython.ipynb
webscapping.ipynb
amazon_webscraping.ipynb
webscapping.ipynb
prime.py
testPython.ipynb
Assign.ipynb
Assignment.ipynb
CO2emissi
Assignment.ipynb > ...
Code
Markdown
Run All
Restart
Clear All Outputs
Variables
Outline
Python 3.12.1
xg.fit(x_train,y_train)
[299]
0.1s
Python
XGBClassifier
XGBClassifier(base_score=None, booster=None, callbacks=None,
colsample_bylevel=None, colsample_bynode=None,
colsample_bytree=None, device=None, early_stopping_rounds=None,
enable_categorical=False, eval_metric=None, feature_types=None,
gamma=None, grow_policy=None, importance_type=None,
interaction_constraints=None, learning_rate=None, max_bin=None,
max_cat_threshold=None, max_cat_to_onehot=None,
max_delta_step=None, max_depth=None, max_leaves=None,
min_child_weight=None, missing=nan, monotone_constraints=None,
multi_strategy=None, n_estimators=None, n_jobs=None,
num_parallel_tree=None, random_state=None, ...)
y_pred_rf=rf.predict(x_test)
y_pred_ad=ad.predict(x_test)
y_pred_gd=gd.predict(x_test)
y_pred_xg=xg.predict(x_test)
[300]
0.0s
Python
from sklearn.metrics import accuracy_score
[301]
0.0s
Python
OUTLINE
TIMELINE
Spaces: 4
Cell 88 of 88
Go Live
Type here to search
ENG
IN
11:08 PM
7/23/2024


File Edit Selection View Go Run
Python
EXPLORER
PYTHON
amazon_webscraping...
Assign.ipynb
Assignment.ipynb
FileHandling.ipynb
flow control.ipynb
Functions.ipynb
Ice-scream.ipynb
Introduction.ipynb
loops.ipynb
oops.ipynb
Pattterns.ipynb
prime.py
testPython.ipynb
webscapping.ipynb
amazon_webscraping.ipynb
webscapping.ipynb
prime.py
testPython.ipynb
Assign.ipynb
Assignment.ipynb
CO2emissi
Assignment.ipynb > ...
Code
Markdown
Run All
Restart
Clear All Outputs
Variables
Outline
Python 3.12.1
y_pred_rf=rf.predict(x_test)
y_pred_ad=ad.predict(x_test)
y_pred_gd=gd.predict(x_test)
y_pred_xg=xg.predict(x_test)
[300]
0.0s
Python
from sklearn.metrics import accuracy_score
[301]
0.0s
Python
print('accuracy_score of rf=',accuracy_score(y_pred_rf,y_test))
print('accuracy_score of ad=',accuracy_score(y_pred_ad,y_test))
print('accuracy_score of gd=',accuracy_score(y_pred_gd,y_test))
print('accuracy_score of xg=',accuracy_score(y_pred_xg,y_test))
[302]
0.0s
Python
accuracy_score of rf= 0.9649122807017544
accuracy_score of ad= 0.9473684210526315
accuracy_score of gd= 0.9473684210526315
accuracy_score of xg= 0.956140350877193
[ ]
Python
OUTLINE
TIMELINE
Spaces: 4
Cell 88 of 88
Go Live
Type here to search
ENG
IN
11:08 PM
7/23/2024