

Output screenshots

Arima output

ritwik@ritwik-VirtualBox: ~/py/ritwik

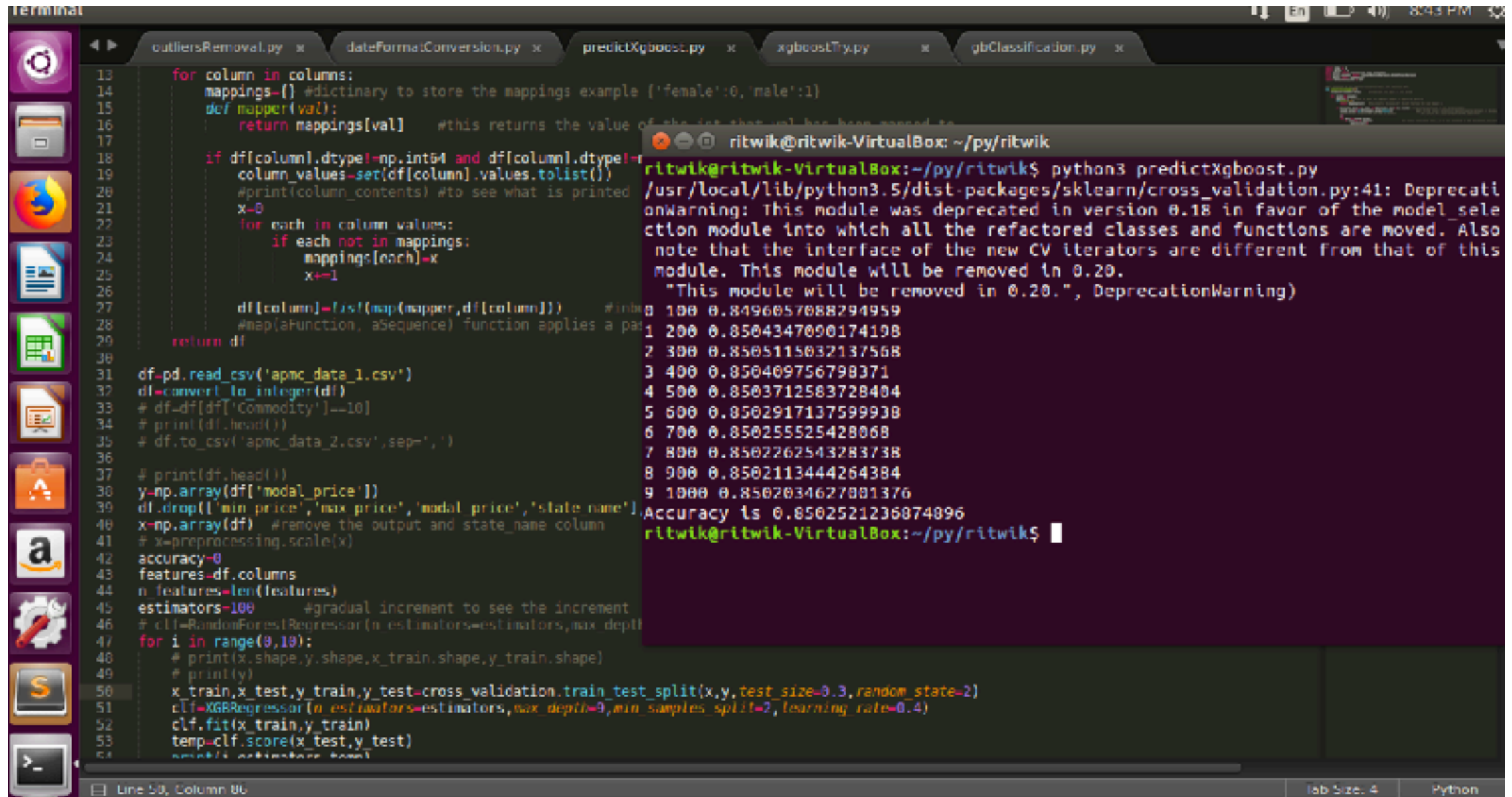
6:26 PM

```
2015-07-01 00:00:00      7.64969
2015-08-01 00:00:00      7.64969
2015-09-01 00:00:00      7.64969
2015-10-01 00:00:00      7.64969
2015-11-01 00:00:00      7.64969
2015-12-01 00:00:00      7.64969
2016-01-01 00:00:00      7.64969
2016-02-01 00:00:00      7.64969
2016-03-01 00:00:00      7.64969
2016-04-01 00:00:00      7.64969
2016-05-01 00:00:00      7.64969
2016-06-01 00:00:00      7.64969
2016-08-01 00:00:00      7.64969
2016-09-01 00:00:00      7.64969
2016-10-01 00:00:00      7.64969
2016-11-01 00:00:00      7.64969
0      [7.662551604934205]
1      [15.26972815831247]
2      [22.870064602841662]
dtype: object
2014-09-01      7.64969
2014-11-01      7.64969
2015-01-01      7.64969
2015-02-01      7.64969
2015-03-01      7.64969
dtype: object
arima_log: 7.649692623711514
2100.0
arima_log: 7.649692623711514
2100.0
arima_log: 7.649692623711514
2100.0
Final prediction(Without scaling up) are: 0      [7.662551604934205]
1      [7.607176553378265]
2      [7.6003364445291925]
dtype: object
Final predictions(After scaling them): [2100.0, 2100.0, 2100.0]
ritwik@ritwik-VirtualBox:~/py/ritwik$
```

XGBoost predict(With outliers removal)

```
ritwik@ritwik-VirtualBox: ~/py/ritwik
ybean': 167, 'lentil': 98, 'nagali': 116, 'small gourd': 163, 'green-peas': 71, 'math (bhaji)': 107, 'nachani': 115, 'snake gourd': 164, 'methi (bhaji)': 110, 'oth.split pulses': 123, 'split black gram': 171, 'turmeric': 188, 'melon': 109, 'spilt pigeon pea': 169, 'goosefoot': 64, 'dhense': 46, 'sweet potato': 179, 'other spices': 127, 'elephant root': 48, 'shahale': 155, 'pigen-pea (bhaji)': 137, 'gulchadi': 75, 'brinjal': 21, 'wal bhaji': 189, 'bullack': 23, 'spinach': 170, 'radish': 147, 'male goat': 103, 'split lentil': 173, 'potato': 143, 'lang': 95, 'betelnuts': 14, 'fodder': 54, 'ghee': 57, 'awala': 6, 'jack fruit': 84, 'garlic': 56, 'pavtta': 135, 'kharbuj': 93, 'guava': 74, 'coconut': 36, 'naspatti': 117, 'amba koy': 0, 'green chilli': 68, 'cluster bean': 35, 'arvi': 4, 'nolkol': 120, 'kagda': 88, 'water melon': 192, 'coriander': 37, 'sarsav': 153, 'pineapple': 139, 'wheat(husked)': 193, 'fenugreek': 51, 'soup berries': 166, 'pavata': 134, 'banana(raw)': 9, 'maize': 100, 'carrot': 27, 'harbara(pendi)': 78, 'black gram': 18, 'flower': 53, 'shepu': 159, 'chandani': 29, 'niger-seed': 119, 'jack fruit(raw)': 85, 'farshi': 49, 'sugarcane': 177, 'custard apple': 45, 'other vegetables': 128, 'kanda pat': 90, 'hilda': 81, 'tag': 180, 'fennel': 50, 'bedana': 12, 'split gram': 172, 'batbati': 11, 'tomato': 186, 'peer': 136, 'shewanti': 161, 'coriander (dry)': 39, 'wood apple': 195, 'wheat(unhusked)': 194, 'walvad': 191, 'ridge gourd': 151, 'bijli': 16, 'pomegranate': 142, 'bhagar/vari': 15, 'chavli (shenga)': 31, 'pumpkin': 144, 'ambat chuka': 1, 'raddish': 146, 'horse gram': 82, 'hemp-seed': 80, 'cowpea': 42, 'aster': 5, 'kardai': 91, 'jambhul': 87, 'tamarind': 181, 'pigeon pea (tur)': 138, 'other cereals': 124, 'onion': 121, 'mango': 105, 'ghevda': 58, 'sheep': 157, 'bitter gourd': 17, 'wal papdi': 190, 'paddy-unhusked': 129, 'chikoo': 32, 'karvand': 92, 'rajgira': 148, 'maize(corn)': 102, 'buffalo': 22, 'matki': 108, 'kand': 89, 'gram': 66, 'hemp': 79, 'maize (corn)': 101, 'cow': 41, 'mango(raw)': 106, 'capsicum': 26, 'sunflower': 178, 'fig': 52, 'linseed': 99, 'tuljapuri': 187, 'strawberi': 175, 'chillies(red)': 33, 'guvar': 76, 'chavli (pala)': 30, 'rala': 149, 'mosambi': 112, 'pappaya (bhaji)': 132, 'sorgum(jawar)': 165, 'mint': 111, 'goats': 63, 'ginger (dry)': 61, 'ginger (fresh)': 62, 'baru seed': 10, 'other oil seeds': 125, 'lemon': 97, 'papnas': 131, 'gr.nut kernels': 65, 'bajri': 7, 'jaggery': 86, 'french bean': 55, 'bullock heart': 24, 'rice(paddy-hus)': 150, 'cucumber': 43, 'mula shenga': 113, 'bottle gourd': 20, 'curry leaves': 44, 'ghosali(bhaji)': 60, 'cotton': 40, 'neem-seed': 118, 'sugar': 176, 'male lamb': 104, 'spilt gerrn gram': 168, 'coriander': 38, 'harbara(bhaji)': 77, 'double bee': 47, 'orange': 122, 'sesamum': 154, 'banana': 8, 'papai': 130, 'squash gourd': 174, 'shevga': 160, 'groundnut pods (wet)': 73, 'skin & bones': 162}
*****
Accuracy is: 0.915211689030037
Enter the APMC code corresponding to name to get the prediction values: 172
Enter the commo code corresponding to name to get the prediction values: 4
Input for prediction: [[ 172    4 2016    12    27    15]]
Output for prediction for future 0 month: [2969.5713]
Input for prediction: [[ 172    4 2017     1    28    15]]
Output for prediction for future 1 month: [2937.0383]
Input for prediction: [[ 172    4 2017     2    29    15]]
Output for prediction for future 2 month: [2969.5713]
{'min_samples_split': 2, 'max_delta_step': 0, 'booster': 'gbtree', 'missing': None, 'random_state': 0, 'n_estimators': 100, 'min_child_weight': 1, 'colsample_bylevel': 1, 'n_jobs': 1, 'subsample': 1, 'seed': None, 'colsample_bytree': 1, 'silent': True, 'reg_alpha': 0, 'learning_rate': 0.5, 'max_depth': 9, 'gamma': 0, 'reg_lambda': 1, 'objective': 'reg:linear', 'scale_pos_weight': 1, 'base_score': 0.5, 'nthread': None}
ritwik@ritwik-VirtualBox: ~/py/ritwik$
```


xgboost(Without outliers)

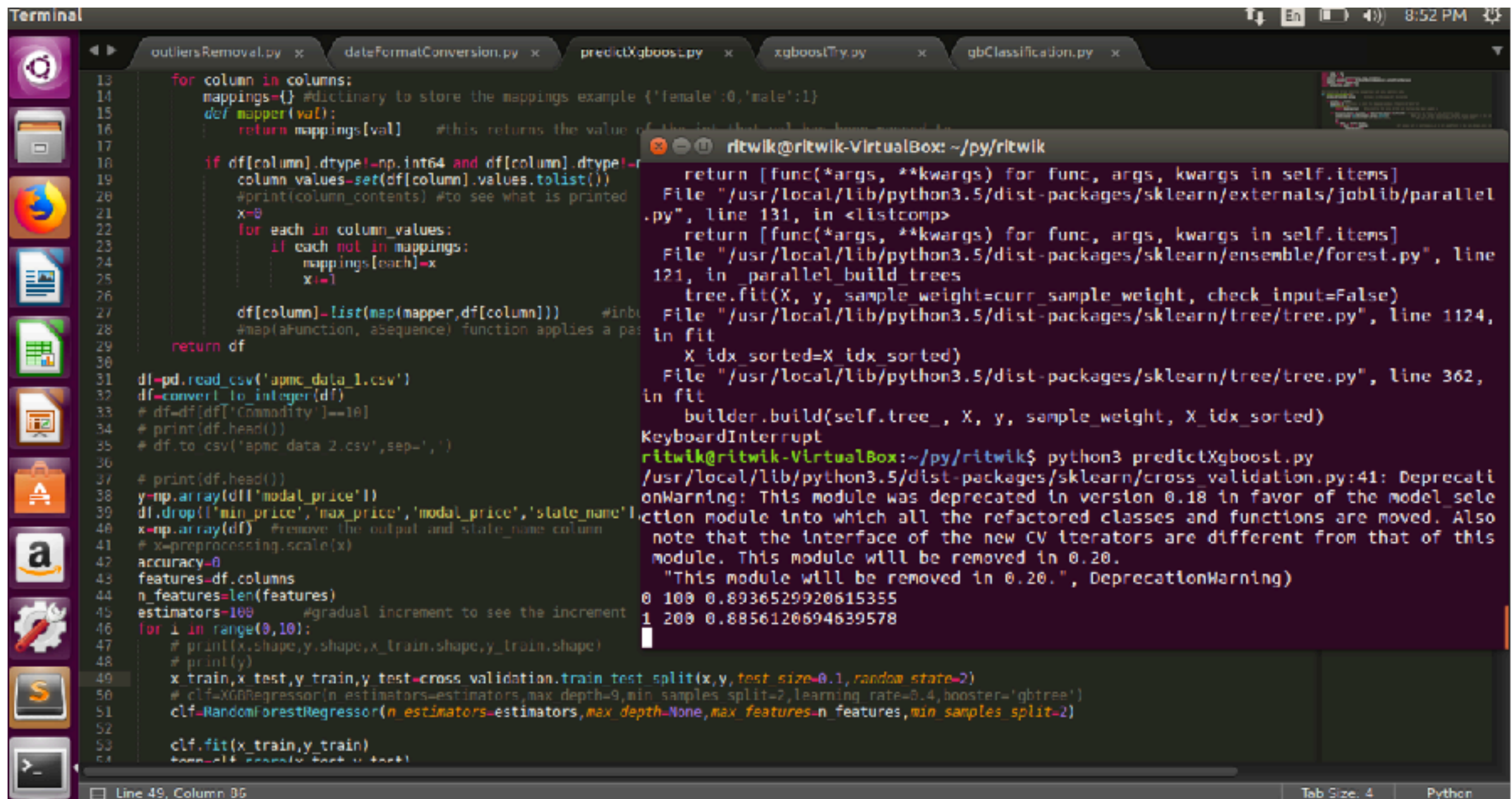


The image shows a code editor with several Python files open: `outliersRemoval.py`, `dateFormatConversion.py`, `predictXgboost.py`, `xgboostTry.py`, and `gbClassification.py`. The `predictXgboost.py` file is the active editor, showing a script for data preprocessing and model training. The script reads a CSV file, converts data to integers, drops certain columns, and uses cross-validation to train an XGBoost model. A terminal window in the foreground shows the command `python3 predictXgboost.py` being executed, displaying a deprecation warning and the output of the model's performance metrics.

```
13 for column in columns:
14     mappings={} #dictionary to store the mappings example {'female':0,'male':1}
15     def mapper(val):
16         return mappings[val] #this returns the value of the dict that val has been mapped to
17
18     if df[column].dtype!=np.int64 and df[column].dtype!=np.int32:
19         column_values=set(df[column].values.tolist())
20         #print(column_values) #to see what is printed
21         x=0
22         for each in column_values:
23             if each not in mappings:
24                 mappings[each]=x
25                 x+=1
26
27     df[column]=list(map(mapper,df[column])) #map
28     #map(aFunction, aSequence) function applies a function to each element of a sequence
29
30 return df
31
32 df=pd.read_csv('apmc_data_1.csv')
33 df=df.convert_objects(convert_numeric=True)
34 # df=df[df['Commodity']!=10]
35 # print(df.head())
36 # df.to_csv('apmc_data_2.csv',sep=',')
37 # print(df.head())
38 y=np.array(df['modal_price'])
39 df.drop(['min price','max price','modal price','state name'],axis=1,inplace=True)
40 x=np.array(df) #remove the output and state_name column
41 # x=preprocessing.scale(x)
42 accuracy=0
43 features=df.columns
44 n_features=len(features)
45 estimators=100 #gradual increment to see the increment
46 # clf=RandomForestRegressor(n_estimators=estimators,max_depth=10)
47 for i in range(0,10):
48     # print(x.shape,y.shape,x_train.shape,y_train.shape)
49     # print(y)
50     x_train,x_test,y_train,y_test=cross_validation.train_test_split(x,y,test_size=0.3,random_state=2)
51     clf=XGBRegressor(n_estimators=estimators,max_depth=9,min_samples_split=2,learning_rate=0.4)
52     clf.fit(x_train,y_train)
53     temp=clf.score(x_test,y_test)
54     #make estimators loop
```

```
ritwik@ritwik-VirtualBox: ~/py/ritwik
ritwik@ritwik-VirtualBox:~/py/ritwik$ python3 predictXgboost.py
/usr/local/lib/python3.5/dist-packages/sklearn/cross_validation.py:41: DeprecationWarning: This module was deprecated in version 0.18 in favor of the model_selection module into which all the refactored classes and functions are moved. Also note that the interface of the new CV iterators are different from that of this module. This module will be removed in 0.20.
  "This module will be removed in 0.20.", DeprecationWarning)
0 100 0.8496057088294959
1 200 0.8504347090174198
2 300 0.8505115032137568
3 400 0.850409756798371
4 500 0.8503712583728404
5 600 0.8502917137599938
6 700 0.85025525428068
7 800 0.8502262543283738
8 900 0.8502113444264384
9 1000 0.8502034627001376
Accuracy is 0.8502521236874896
ritwik@ritwik-VirtualBox:~/py/ritwik$
```

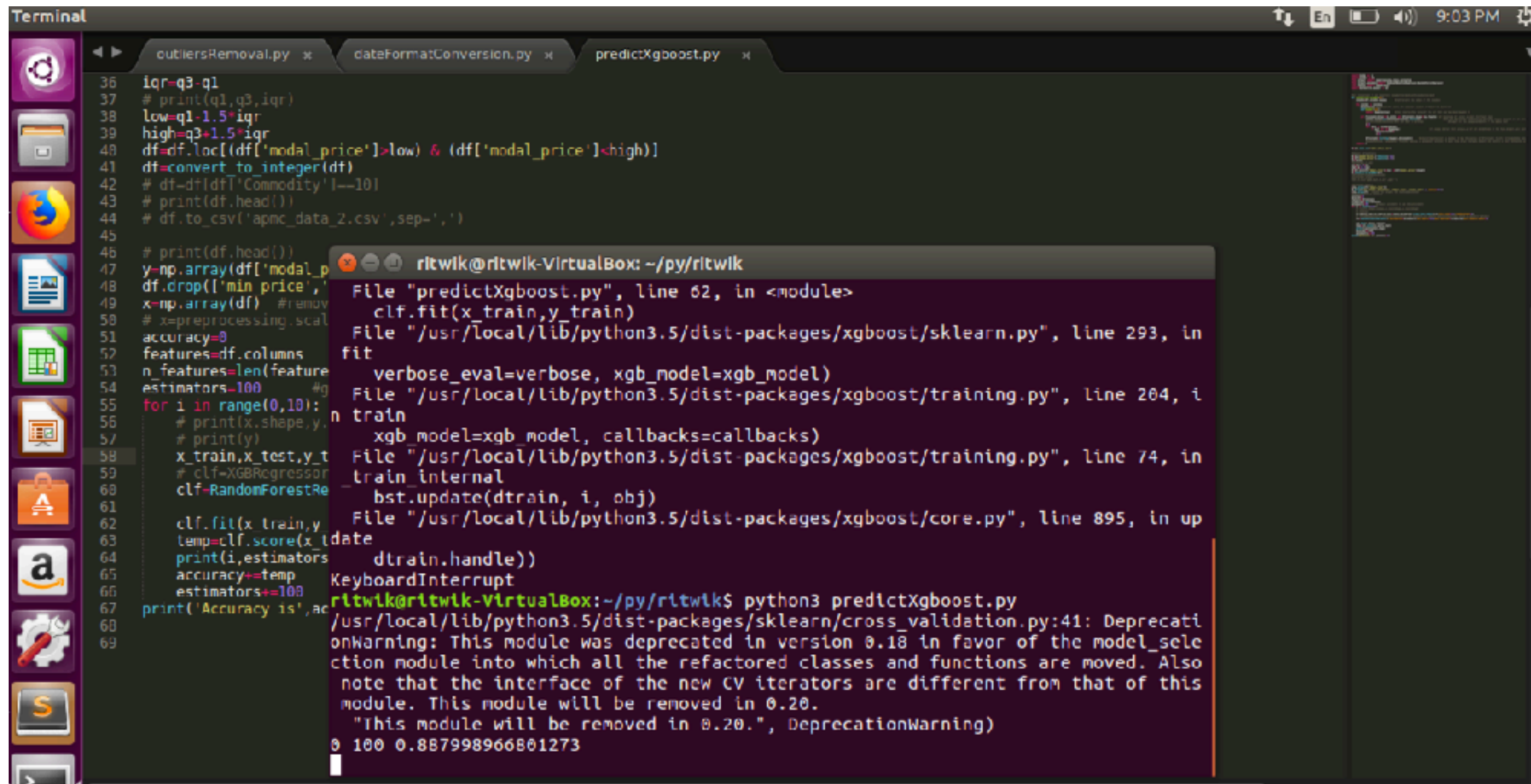
Random forest regressor (With outliers removal)



```
Terminal
outliersRemoval.py x dateFormatConversion.py x predictXgboost.py x xgboostTry.py x qbClassification.py x

13 for column in columns:
14     mappings={} #dictionary to store the mappings example {'female':0,'male':1}
15     def mapper(val):
16         return mappings[val] #this returns the value of the int that val has been mapped to
17
18     if df[column].dtype!=np.int64 and df[column].dtype!=
19         column values=set(df[column].values.tolist())
20         #print(column_contents) #to see what is printed
21         x=0
22         for each in column_values:
23             if each not in mappings:
24                 mappings[each]=x
25                 x+=1
26
27     df[column]=list(map(mapper,df[column])) #inbu
28     #map(afunction, asequence) function applies a pas
29     return df
30
31 df=pd.read_csv('apmc_data_1.csv')
32 df=convert_to_integer(df)
33 # df=df[df['Commodity']==10]
34 # print(df.head())
35 # df.to_csv('apmc_data_2.csv',sep=',')
36
37 # print(df.head())
38 y=np.array(df['modal_price'])
39 df.drop(['min_price','max_price','modal_price','state_name'])
40 x=np.array(df) #remove the output and state_name column
41 # x=preprocessing.scale(x)
42 accuracy=0
43 features=df.columns
44 n_features=len(features)
45 estimators=100 #gradual increment to see the increment
46 for i in range(0,10):
47     # print(x.shape,y.shape,x_train.shape,y_train.shape)
48     # print(y)
49     x_train,x_test,y_train,y_test=cross_validation.train_test_split(x,y,test_size=0.1,random_state=2)
50     # clf=XGBRegressor(n_estimators=estimators,max_depth=9,min_samples_split=2,learning_rate=0.4,booster='gbtree')
51     clf=RandomForestRegressor(n_estimators=estimators,max_depth=None,max_features=n_features,min_samples_split=2)
52
53     clf.fit(x_train,y_train)
54     temp=clf.predict(x_test)
55
rtwik@ritwik-VirtualBox: ~/py/ritwik
return [func(*args, **kwargs) for func, args, kwargs in self.items]
File "/usr/local/lib/python3.5/dist-packages/sklearn/externals/joblib/parallel
.py", line 131, in <listcomp>
return [func(*args, **kwargs) for func, args, kwargs in self.items]
File "/usr/local/lib/python3.5/dist-packages/sklearn/ensemble/forest.py", line
121, in _parallel_build_trees
tree.fit(X, y, sample_weight=curr_sample_weight, check_input=False)
File "/usr/local/lib/python3.5/dist-packages/sklearn/tree/tree.py", line 1124,
in fit
X_idx_sorted=X_idx_sorted)
File "/usr/local/lib/python3.5/dist-packages/sklearn/tree/tree.py", line 362,
in fit
builder.build(self.tree_, X, y, sample_weight, X_idx_sorted)
KeyboardInterrupt
rtwik@ritwik-VirtualBox:~/py/ritwik$ python3 predictXgboost.py
/usr/local/lib/python3.5/dist-packages/sklearn/cross_validation.py:41: Deprecati
onWarning: This module was deprecated in version 0.18 in favor of the model_sele
ction module into which all the refactored classes and functions are moved. Also
note that the interface of the new CV iterators are different from that of this
module. This module will be removed in 0.20.
"This module will be removed in 0.20.", DeprecationWarning)
0 100 0.8936529920615355
1 200 0.8856120694639578
```


Random forest regressor (Without outliers removal)



```
Terminal
outliersRemoval.py x dateFormatConversion.py x predictXgboost.py x

36 iqr=q3-q1
37 # print(q1,q3,iqr)
38 low=q1-1.5*iqr
39 high=q3+1.5*iqr
40 df=df.loc[(df['modal_price']>low) & (df['modal_price']<high)]
41 df=convert_to_integer(df)
42 # df=df[df['Commodity']!=10]
43 # print(df.head())
44 # df.to_csv('apmc_data_2.csv',sep=',')
45
46 # print(df.head())
47 y=np.array(df['modal_p
48 df.drop(['min price',
49 x=np.array(df) #remov
50 # x=preprocessing.scal
51 accuracy=0
52 features=df.columns
53 n_features=len(feature
54 estimators=100 #g
55 for i in range(0,10):
56     # print(x.shape,y.
57     # print(y)
58     x_train,x_test,y_t
59     # clf=XGBRegressor
60     clf=RandomForestRe
61
62     clf.fit(x_train,y
63     temp=clf.score(x_t
64     print(i,estimators
65     accuracy+=temp
66     estimators+=100
67 print('Accuracy is',ac
68
69

ritwik@ritwik-VirtualBox: ~/py/ritwik
File "predictXgboost.py", line 62, in <module>
    clf.fit(x_train,y_train)
File "/usr/local/lib/python3.5/dist-packages/xgboost/sklearn.py", line 293, in
fit
    verbose_eval=verbose, xgb_model=xgb_model)
File "/usr/local/lib/python3.5/dist-packages/xgboost/training.py", line 204, i
n train
    xgb_model=xgb_model, callbacks=callbacks)
File "/usr/local/lib/python3.5/dist-packages/xgboost/training.py", line 74, in
_train_internal
    bst.update(dtrain, i, obj)
File "/usr/local/lib/python3.5/dist-packages/xgboost/core.py", line 895, in up
KeyboardInterrupt
ritwik@ritwik-VirtualBox: ~/py/ritwik$ python3 predictXgboost.py
/usr/local/lib/python3.5/dist-packages/sklearn/cross_validation.py:41: Deprecati
onWarning: This module was deprecated in version 0.18 in favor of the model_sele
ction module into which all the refactored classes and functions are moved. Also
note that the interface of the new CV iterators are different from that of this
module. This module will be removed in 0.20.
  "This module will be removed in 0.20.", DeprecationWarning)
0 100 0.887998966801273
```