xgboost

May 29, 2023

1 PREDICTION WITH XGBRegressor

[37]:
```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from math import sqrt
from sklearn.metrics import mean_squared_error
from xgboost import XGBRegressor
```

[4]:
```
# read dataset may2023
df = pd.read_pickle("../../data/20230319_RTU_Dataset_PPC-Lab/combined_may2023.
↪pkl")
```

[6]:
```
df
```

[6]:

	MEM_USAGE	CPU_USAGE	PS1_V	TEMP
0	35.555417	27.343750	5.435294	28.687
1	35.555417	6.367041	5.435294	28.687
2	35.555417	7.142857	5.435294	28.687
3	35.555417	27.306273	5.435294	28.687
4	35.555417	5.639098	5.435294	28.687
…	…	…	…	…
3798	25.962425	8.396947	5.383530	29.562
3799	25.962425	6.766917	5.383530	29.562
3800	25.962425	6.000000	5.383530	29.562
3801	25.962425	8.045977	5.383530	29.562
3802	25.962425	13.229572	5.383530	29.562

[3733 rows x 4 columns]

[7]:
```
def mean_absolute_percentage_error(y_true, y_pred):
    y_true, y_pred = np.array(y_true), np.array(y_pred)
    return np.mean(np.abs((y_true - y_pred) / y_true)) * 100
```

[21]:
```
training_size = int(len(df) * 0.7)
validation_size = int(len(df) * 0.8)
```

```
x_train = [[i] for i in df["TEMP"]][:training_size]
y_train = [i for i in df["CPU_USAGE"]][:training_size]

x_val = [[i] for i in df["TEMP"]][training_size:validation_size]
y_val = [i for i in df["CPU_USAGE"]][training_size:validation_size]

x_test = [[i] for i in df["TEMP"]][validation_size:]
y_test = [[i] for i in df["CPU_USAGE"]][validation_size:]

len(x_test)
```

[21]: 747

[25]:
```
# Training
regressor = XGBRegressor(
    max_depth=10,
    n_estimators=1000,
    min_child_weight=0.5,
    colsample_bytree=0.8,
    subsample=0.8,
    eta=0.1,
    seed=42)
```

[26]:
```
regressor.fit(
    x_train,
    y_train,
    eval_metric="rmse",
    eval_set=[(x_train, y_train), (x_val, y_val)],
    verbose=True,
    early_stopping_rounds = 20)
```

```
[0]     validation_0-rmse:16.30337      validation_1-rmse:17.70207
[1]     validation_0-rmse:15.42617      validation_1-rmse:16.92710
[2]     validation_0-rmse:14.67558      validation_1-rmse:16.25013
[3]     validation_0-rmse:14.04135      validation_1-rmse:15.65830
[4]     validation_0-rmse:13.51214      validation_1-rmse:15.16591
[5]     validation_0-rmse:13.05991      validation_1-rmse:14.77737
[6]     validation_0-rmse:12.68998      validation_1-rmse:14.43652
[7]     validation_0-rmse:12.37387      validation_1-rmse:14.15391
[8]     validation_0-rmse:12.11370      validation_1-rmse:13.91361
[9]     validation_0-rmse:11.89484      validation_1-rmse:13.71257
[10]    validation_0-rmse:11.71473      validation_1-rmse:13.55021
[11]    validation_0-rmse:11.57144      validation_1-rmse:13.41213
[12]    validation_0-rmse:11.45046      validation_1-rmse:13.29746
[13]    validation_0-rmse:11.35357      validation_1-rmse:13.21299
[14]    validation_0-rmse:11.26985      validation_1-rmse:13.12962
```

```
[15]    validation_0-rmse:11.20182      validation_1-rmse:13.07147
[16]    validation_0-rmse:11.15427      validation_1-rmse:13.02788
[17]    validation_0-rmse:11.11023      validation_1-rmse:12.98859
[18]    validation_0-rmse:11.07286      validation_1-rmse:12.94716
[19]    validation_0-rmse:11.04300      validation_1-rmse:12.91648
[20]    validation_0-rmse:11.02035      validation_1-rmse:12.89088
[21]    validation_0-rmse:11.00103      validation_1-rmse:12.86950
[22]    validation_0-rmse:10.98545      validation_1-rmse:12.84599
[23]    validation_0-rmse:10.97124      validation_1-rmse:12.83003
[24]    validation_0-rmse:10.96151      validation_1-rmse:12.81423
[25]    validation_0-rmse:10.95283      validation_1-rmse:12.80021
[26]    validation_0-rmse:10.94656      validation_1-rmse:12.79633
[27]    validation_0-rmse:10.94175      validation_1-rmse:12.78995
[28]    validation_0-rmse:10.93800      validation_1-rmse:12.78480
[29]    validation_0-rmse:10.93368      validation_1-rmse:12.77937
[30]    validation_0-rmse:10.93125      validation_1-rmse:12.77834
[31]    validation_0-rmse:10.92900      validation_1-rmse:12.77148
[32]    validation_0-rmse:10.92634      validation_1-rmse:12.77154
[33]    validation_0-rmse:10.92398      validation_1-rmse:12.76382
[34]    validation_0-rmse:10.92233      validation_1-rmse:12.75696
[35]    validation_0-rmse:10.92172      validation_1-rmse:12.75781
[36]    validation_0-rmse:10.92032      validation_1-rmse:12.75240
[37]    validation_0-rmse:10.91964      validation_1-rmse:12.75079
[38]    validation_0-rmse:10.91918      validation_1-rmse:12.74785
[39]    validation_0-rmse:10.91868      validation_1-rmse:12.74641
[40]    validation_0-rmse:10.91819      validation_1-rmse:12.75077
[41]    validation_0-rmse:10.91802      validation_1-rmse:12.74825
[42]    validation_0-rmse:10.91801      validation_1-rmse:12.74851
[43]    validation_0-rmse:10.91774      validation_1-rmse:12.74517
[44]    validation_0-rmse:10.91753      validation_1-rmse:12.74387
[45]    validation_0-rmse:10.91741      validation_1-rmse:12.74279
[46]    validation_0-rmse:10.91745      validation_1-rmse:12.74146
[47]    validation_0-rmse:10.91731      validation_1-rmse:12.74144
[48]    validation_0-rmse:10.91747      validation_1-rmse:12.73917
[49]    validation_0-rmse:10.91747      validation_1-rmse:12.73924
[50]    validation_0-rmse:10.91748      validation_1-rmse:12.73812
[51]    validation_0-rmse:10.91737      validation_1-rmse:12.73659
[52]    validation_0-rmse:10.91733      validation_1-rmse:12.74073
[53]    validation_0-rmse:10.91719      validation_1-rmse:12.74025
[54]    validation_0-rmse:10.91722      validation_1-rmse:12.73851
[55]    validation_0-rmse:10.91711      validation_1-rmse:12.73988
[56]    validation_0-rmse:10.91704      validation_1-rmse:12.73897
[57]    validation_0-rmse:10.91698      validation_1-rmse:12.74090
[58]    validation_0-rmse:10.91689      validation_1-rmse:12.73768
[59]    validation_0-rmse:10.91683      validation_1-rmse:12.74057
[60]    validation_0-rmse:10.91696      validation_1-rmse:12.73805
[61]    validation_0-rmse:10.91692      validation_1-rmse:12.73692
[62]    validation_0-rmse:10.91699      validation_1-rmse:12.73839
```

```
[63]    validation_0-rmse:10.91689      validation_1-rmse:12.73574
[64]    validation_0-rmse:10.91687      validation_1-rmse:12.73674
[65]    validation_0-rmse:10.91679      validation_1-rmse:12.73929
[66]    validation_0-rmse:10.91679      validation_1-rmse:12.73919
[67]    validation_0-rmse:10.91675      validation_1-rmse:12.73780
[68]    validation_0-rmse:10.91675      validation_1-rmse:12.73701
[69]    validation_0-rmse:10.91681      validation_1-rmse:12.73766
[70]    validation_0-rmse:10.91678      validation_1-rmse:12.73676
[71]    validation_0-rmse:10.91677      validation_1-rmse:12.73500
[72]    validation_0-rmse:10.91678      validation_1-rmse:12.73236
[73]    validation_0-rmse:10.91677      validation_1-rmse:12.73301
[74]    validation_0-rmse:10.91681      validation_1-rmse:12.73325
[75]    validation_0-rmse:10.91691      validation_1-rmse:12.73056
[76]    validation_0-rmse:10.91689      validation_1-rmse:12.72969
[77]    validation_0-rmse:10.91688      validation_1-rmse:12.73411
[78]    validation_0-rmse:10.91686      validation_1-rmse:12.73107
[79]    validation_0-rmse:10.91679      validation_1-rmse:12.73619
[80]    validation_0-rmse:10.91677      validation_1-rmse:12.73680
[81]    validation_0-rmse:10.91675      validation_1-rmse:12.73557
[82]    validation_0-rmse:10.91678      validation_1-rmse:12.73518
[83]    validation_0-rmse:10.91678      validation_1-rmse:12.73343
[84]    validation_0-rmse:10.91676      validation_1-rmse:12.73407
[85]    validation_0-rmse:10.91677      validation_1-rmse:12.73748
[86]    validation_0-rmse:10.91681      validation_1-rmse:12.73344
[87]    validation_0-rmse:10.91682      validation_1-rmse:12.73203
[88]    validation_0-rmse:10.91687      validation_1-rmse:12.73322
[89]    validation_0-rmse:10.91678      validation_1-rmse:12.73616
[90]    validation_0-rmse:10.91678      validation_1-rmse:12.73527
[91]    validation_0-rmse:10.91678      validation_1-rmse:12.73812
[92]    validation_0-rmse:10.91679      validation_1-rmse:12.73472
[93]    validation_0-rmse:10.91679      validation_1-rmse:12.73241
[94]    validation_0-rmse:10.91679      validation_1-rmse:12.73683
[95]    validation_0-rmse:10.91688      validation_1-rmse:12.73931
[96]    validation_0-rmse:10.91691      validation_1-rmse:12.73699
```

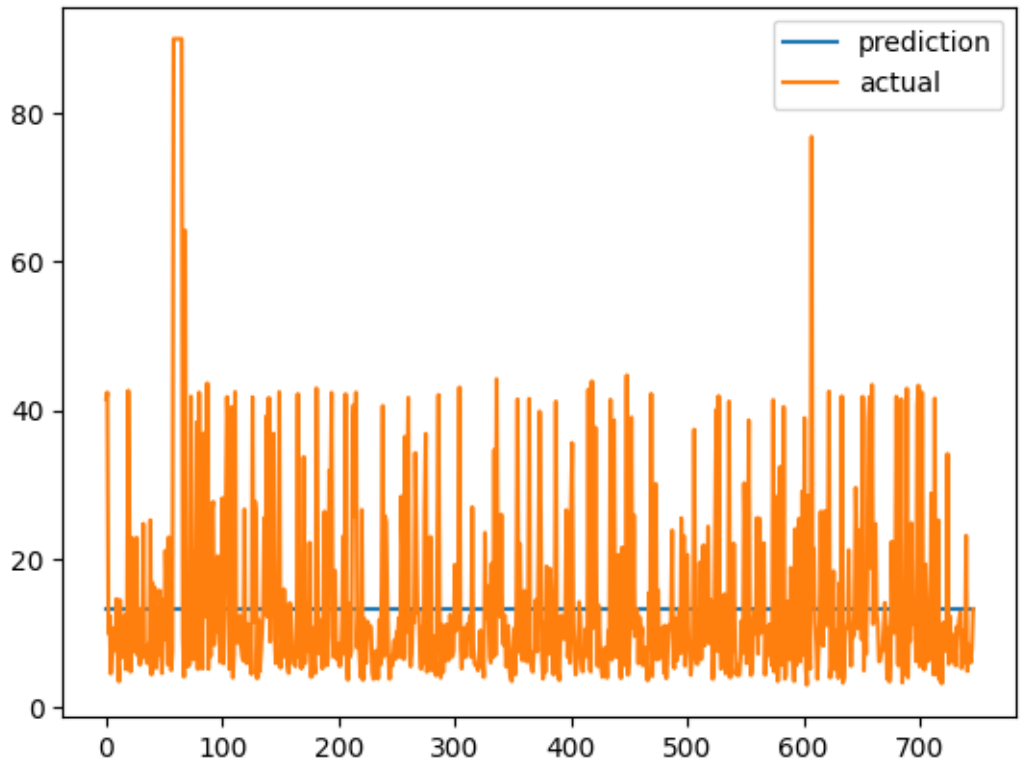
[26]:
```
XGBRegressor(base_score=None, booster=None, callbacks=None,
             colsample_bylevel=None, colsample_bynode=None,
             colsample_bytree=0.8, early_stopping_rounds=None,
             enable_categorical=False, eta=0.1, eval_metric=None,
             feature_types=None, gamma=None, gpu_id=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=10,
             max_leaves=None, min_child_weight=0.5, missing=nan,
             monotone_constraints=None, n_estimators=1000, n_jobs=None,
             num_parallel_tree=None, predictor=None, …)
```

[28]:
```
Y_pred = regressor.predict(x_test)
```

[29]:
```
print(mean_absolute_percentage_error(list(Y_pred), y_test))
```

66.18822223099487

[30]:
```
import matplotlib.pyplot as plt
import numpy as np

x = range(len(list(Y_pred)))
y_pred = list(Y_pred)
y_actual = y_test

plt.plot(x, y_pred, label="prediction")
plt.plot(x, y_actual, label="actual")
plt.legend()
plt.show()
```



[31]:
```
training_size = int(len(df) * 0.7)
validation_size = int(len(df) * 0.8)

x_train = [[i] for i in df["CPU_USAGE"]][:training_size]
```

```
y_train = [i for i in df["TEMP"]][:training_size]

x_val = [[i] for i in df["CPU_USAGE"]][training_size:validation_size]
y_val = [i for i in df["TEMP"]][training_size:validation_size]

x_test = [[i] for i in df["CPU_USAGE"]][validation_size:]
y_test = [[i] for i in df["TEMP"]][validation_size:]
```

[32]:
```
# Training
regressor = XGBRegressor(
    max_depth=10,
    n_estimators=1000,
    min_child_weight=0.5,
    colsample_bytree=0.8,
    subsample=0.8,
    eta=0.1,
    seed=42)

regressor.fit(
    x_train,
    y_train,
    eval_metric="rmse",
    eval_set=[(x_train, y_train), (x_val, y_val)],
    verbose=True,
    early_stopping_rounds = 20)
```

```
[0]     validation_0-rmse:22.08233      validation_1-rmse:24.63473
[1]     validation_0-rmse:19.92003      validation_1-rmse:22.48453
[2]     validation_0-rmse:17.97698      validation_1-rmse:20.55397
[3]     validation_0-rmse:16.23205      validation_1-rmse:18.82178
[4]     validation_0-rmse:14.66778      validation_1-rmse:17.27032
[5]     validation_0-rmse:13.26756      validation_1-rmse:15.88274
[6]     validation_0-rmse:12.01574      validation_1-rmse:14.65642
[7]     validation_0-rmse:10.89029      validation_1-rmse:13.55449
[8]     validation_0-rmse:9.88926       validation_1-rmse:12.56441
[9]     validation_0-rmse:8.99528       validation_1-rmse:11.69038
[10]    validation_0-rmse:8.20079       validation_1-rmse:10.91656
[11]    validation_0-rmse:7.49468       validation_1-rmse:10.21608
[12]    validation_0-rmse:6.87242       validation_1-rmse:9.60787
[13]    validation_0-rmse:6.32195       validation_1-rmse:9.06987
[14]    validation_0-rmse:5.83451       validation_1-rmse:8.59007
[15]    validation_0-rmse:5.40613       validation_1-rmse:8.15505
[16]    validation_0-rmse:5.03438       validation_1-rmse:7.77932
[17]    validation_0-rmse:4.70938       validation_1-rmse:7.44948
[18]    validation_0-rmse:4.42777       validation_1-rmse:7.17368
[19]    validation_0-rmse:4.17682       validation_1-rmse:6.91071
[20]    validation_0-rmse:3.96330       validation_1-rmse:6.68455
```

```
[21]    validation_0-rmse:3.78039       validation_1-rmse:6.48317
[22]    validation_0-rmse:3.62120       validation_1-rmse:6.31089
[23]    validation_0-rmse:3.48424       validation_1-rmse:6.15673
[24]    validation_0-rmse:3.36571       validation_1-rmse:6.02343
[25]    validation_0-rmse:3.27064       validation_1-rmse:5.91111
[26]    validation_0-rmse:3.18804       validation_1-rmse:5.80869
[27]    validation_0-rmse:3.11149       validation_1-rmse:5.71094
[28]    validation_0-rmse:3.05106       validation_1-rmse:5.63712
[29]    validation_0-rmse:2.99961       validation_1-rmse:5.56599
[30]    validation_0-rmse:2.95543       validation_1-rmse:5.49878
[31]    validation_0-rmse:2.91971       validation_1-rmse:5.44001
[32]    validation_0-rmse:2.88715       validation_1-rmse:5.39038
[33]    validation_0-rmse:2.86409       validation_1-rmse:5.34742
[34]    validation_0-rmse:2.84479       validation_1-rmse:5.30557
[35]    validation_0-rmse:2.82183       validation_1-rmse:5.27172
[36]    validation_0-rmse:2.80003       validation_1-rmse:5.23819
[37]    validation_0-rmse:2.78484       validation_1-rmse:5.21284
[38]    validation_0-rmse:2.77332       validation_1-rmse:5.19096
[39]    validation_0-rmse:2.75964       validation_1-rmse:5.16544
[40]    validation_0-rmse:2.75338       validation_1-rmse:5.14638
[41]    validation_0-rmse:2.74072       validation_1-rmse:5.13245
[42]    validation_0-rmse:2.73115       validation_1-rmse:5.11771
[43]    validation_0-rmse:2.72124       validation_1-rmse:5.10340
[44]    validation_0-rmse:2.71210       validation_1-rmse:5.09186
[45]    validation_0-rmse:2.69885       validation_1-rmse:5.08109
[46]    validation_0-rmse:2.69026       validation_1-rmse:5.07445
[47]    validation_0-rmse:2.68445       validation_1-rmse:5.06615
[48]    validation_0-rmse:2.67685       validation_1-rmse:5.05828
[49]    validation_0-rmse:2.66827       validation_1-rmse:5.05370
[50]    validation_0-rmse:2.66199       validation_1-rmse:5.04774
[51]    validation_0-rmse:2.65569       validation_1-rmse:5.03908
[52]    validation_0-rmse:2.64803       validation_1-rmse:5.03312
[53]    validation_0-rmse:2.64502       validation_1-rmse:5.02758
[54]    validation_0-rmse:2.64103       validation_1-rmse:5.02359
[55]    validation_0-rmse:2.63627       validation_1-rmse:5.01574
[56]    validation_0-rmse:2.62972       validation_1-rmse:5.01291
[57]    validation_0-rmse:2.62640       validation_1-rmse:5.01048
[58]    validation_0-rmse:2.62098       validation_1-rmse:5.00502
[59]    validation_0-rmse:2.61603       validation_1-rmse:5.00120
[60]    validation_0-rmse:2.61217       validation_1-rmse:5.00162
[61]    validation_0-rmse:2.60868       validation_1-rmse:4.99736
[62]    validation_0-rmse:2.60524       validation_1-rmse:5.00019
[63]    validation_0-rmse:2.60108       validation_1-rmse:5.00392
[64]    validation_0-rmse:2.59663       validation_1-rmse:5.00211
[65]    validation_0-rmse:2.59305       validation_1-rmse:5.00317
[66]    validation_0-rmse:2.58995       validation_1-rmse:5.00208
[67]    validation_0-rmse:2.58555       validation_1-rmse:5.00053
[68]    validation_0-rmse:2.57838       validation_1-rmse:5.00326
```

```
[69]    validation_0-rmse:2.57600       validation_1-rmse:5.00455
[70]    validation_0-rmse:2.57357       validation_1-rmse:5.00432
[71]    validation_0-rmse:2.57158       validation_1-rmse:5.00741
[72]    validation_0-rmse:2.57055       validation_1-rmse:5.00902
[73]    validation_0-rmse:2.56886       validation_1-rmse:5.00727
[74]    validation_0-rmse:2.56433       validation_1-rmse:5.00412
[75]    validation_0-rmse:2.56183       validation_1-rmse:5.00716
[76]    validation_0-rmse:2.55721       validation_1-rmse:5.00911
[77]    validation_0-rmse:2.55519       validation_1-rmse:5.01089
[78]    validation_0-rmse:2.55271       validation_1-rmse:5.01084
[79]    validation_0-rmse:2.55161       validation_1-rmse:5.01306
[80]    validation_0-rmse:2.54706       validation_1-rmse:5.01290
[81]    validation_0-rmse:2.54417       validation_1-rmse:5.01236
```

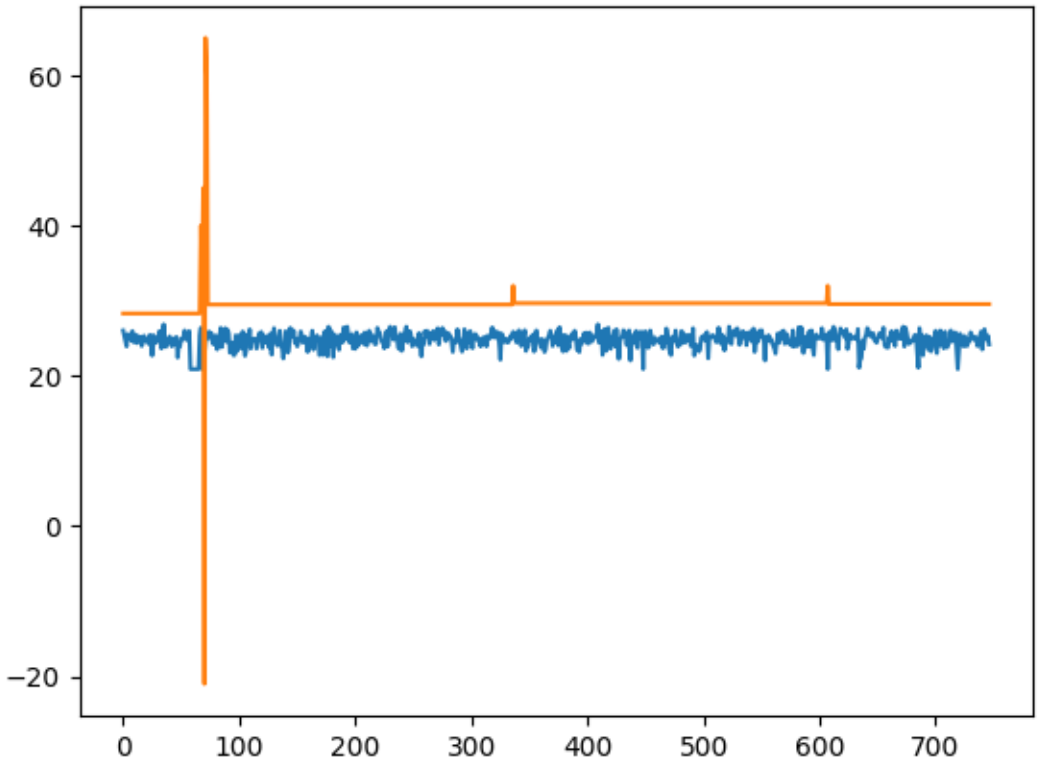
[32]:
```
XGBRegressor(base_score=None, booster=None, callbacks=None,
             colsample_bylevel=None, colsample_bynode=None,
             colsample_bytree=0.8, early_stopping_rounds=None,
             enable_categorical=False, eta=0.1, eval_metric=None,
             feature_types=None, gamma=None, gpu_id=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=10,
             max_leaves=None, min_child_weight=0.5, missing=nan,
             monotone_constraints=None, n_estimators=1000, n_jobs=None,
             num_parallel_tree=None, predictor=None, …)
```

[33]:
```
Y_pred = regressor.predict(x_test)
```

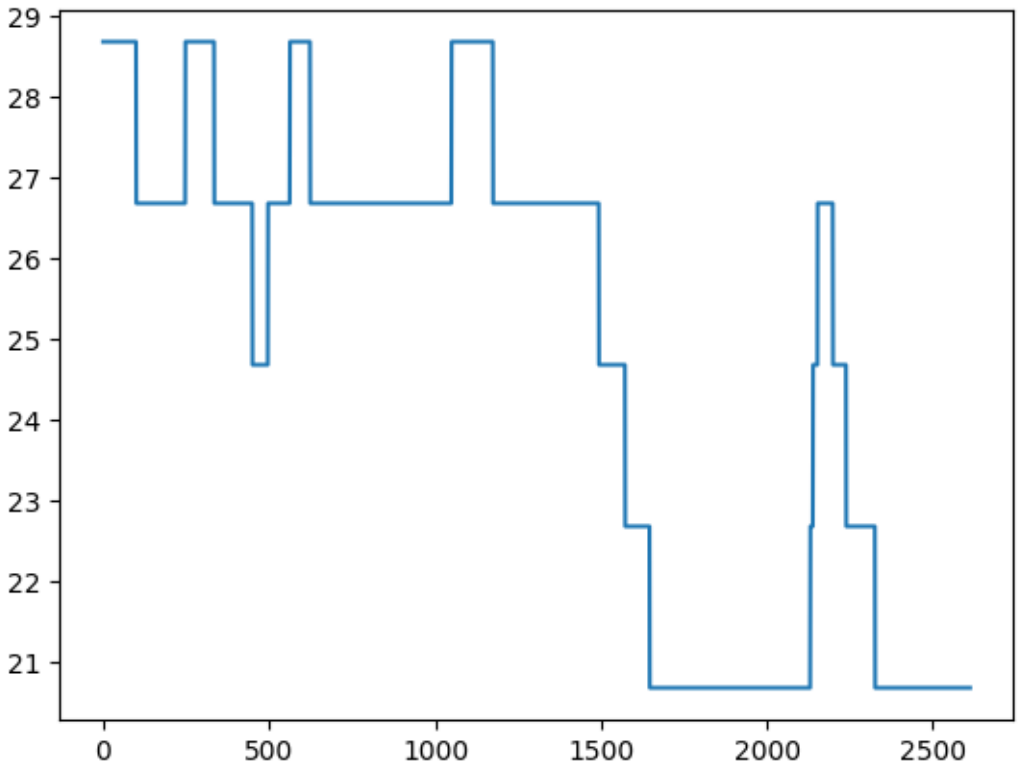
[34]:
```
print(mean_absolute_percentage_error(Y_pred, y_test))
```

```
19.84582274749494
```

[35]:
```
import matplotlib.pyplot as plt
import numpy as np

x = range(len(list(Y_pred)))
y_pred = list(Y_pred)
y_actual = y_test

plt.plot(x, y_pred)
plt.plot(x, y_actual)
plt.show()
```



[36]:
```
x = range(len(list(y_train)))
y_pred = list(y_train)

plt.plot(x, y_pred)
plt.show()
```


29
28
27
26
25
24
23
22
21
0
500
1000
1500
2000
2500