

```
In [4]: import numpy as np
from catboost import Pool, CatBoostRegressor
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error
import pickle
import joblib
```

```
In [5]: air=pd.read_csv("station_day.csv")
```

```
In [6]: df = air
a = df.mean()
df = df.fillna(round(a))
air = df
air
```

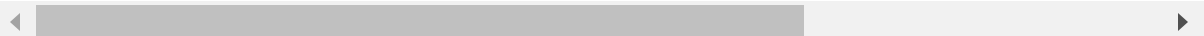
C:\Users\ishan\AppData\Local\Temp\ipykernel_6892\35374477.py:2: FutureWarning: Dropping of nuisance columns in DataFrame reductions (with 'numeric_only=None') is deprecated; in a future version this will raise TypeError. Select only valid columns before calling the reduction.

```
a = df.mean()
```

Out[6]:

	StationId	Date	PM2.5	PM10	NO	NO2	NOx	NH3	CO	SO2	O3	PM10
0	AP001	11/24/2017	71.36	115.75	1.75	20.65	12.40	12.19	0.10	10.76	109.26	115.75
1	AP001	11/25/2017	81.40	124.50	1.44	20.50	12.08	10.72	0.12	15.24	127.09	124.50
2	AP001	11/26/2017	78.32	129.06	1.26	26.00	14.85	10.28	0.14	26.96	117.44	129.06
3	AP001	11/27/2017	88.76	135.32	6.60	30.85	21.77	12.91	0.11	33.59	111.81	135.32
4	AP001	11/28/2017	64.18	104.09	2.56	28.07	17.01	11.42	0.09	19.00	138.18	104.09
...
108030	WB013	6/27/2020	8.65	16.46	23.00	35.00	41.00	29.00	0.69	4.36	30.59	16.46
108031	WB013	6/28/2020	11.80	18.47	23.00	35.00	41.00	29.00	0.68	3.49	38.95	18.47
108032	WB013	6/29/2020	18.60	32.26	13.65	200.87	214.20	11.40	0.78	5.12	38.17	32.26
108033	WB013	6/30/2020	16.07	39.30	7.56	29.13	36.69	29.26	0.69	5.88	29.64	39.30
108034	WB013	7/1/2020	10.50	36.50	7.78	22.50	30.25	27.23	0.58	2.80	13.10	36.50

108035 rows × 16 columns



```
In [7]: X=air[["PM2.5", "PM10", "O3", "NO", "NO2", "NOx", "NH3", "CO", "SO2"]]
y=air["AQI"]
```

```
In [8]: y.shape
```

Out[8]: (108035,)

```
In [9]: X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2)
```

```
In [11]: train_pool = Pool(X_train,
                           y_train,
                           cat_features=None)
test_pool = Pool(X_test,
                 y_test,
                 cat_features=None)
```

```
In [12]: model = CatBoostRegressor(iterations=3000,
                                   depth=10,
                                   learning_rate = 0.01,
                                   loss_function='RMSE',
                                   eval_metric = 'RMSE',
                                   random_seed = 55,
                                   od_type = "Iter",
                                   metric_period = 1,
                                   od_wait = 20)
```

```
In [13]: model.fit(train_pool)
```

	learn:	total:	remaining:
87:	66.5526236	2.85s	1m 34s
88:	66.2170356	2.89s	1m 34s
89:	65.8922714	2.92s	1m 34s
90:	65.5542214	2.96s	1m 34s
91:	65.2214724	2.99s	1m 34s
92:	64.8915032	3.02s	1m 34s
93:	64.5753448	3.05s	1m 34s
94:	64.2731694	3.09s	1m 34s
95:	63.9634424	3.12s	1m 34s
96:	63.6619963	3.15s	1m 34s
97:	63.3499181	3.18s	1m 34s
98:	63.0587428	3.21s	1m 34s
99:	62.7558224	3.25s	1m 34s
100:	62.4600630	3.28s	1m 34s
101:	62.1723573	3.32s	1m 34s
102:	61.8949086	3.35s	1m 34s
103:	61.6080214	3.39s	1m 34s
104:	61.3349669	3.42s	1m 34s
105:	61.0689145	3.45s	1m 34s

```
In [14]: preds = model.predict(test_pool)
```

```
In [15]: print(preds)
```

```
[208.46637072 169.10522518 233.00352397 ... 189.84308526 124.98460981
197.27429728]
```

```
In [16]: preds
```

```
Out[16]: array([208.46637072, 169.10522518, 233.00352397, ..., 189.84308526,
124.98460981, 197.27429728])
```

In [17]: `y_test`

```
Out[17]: 37808      223.0
          22592      180.0
          105299     214.0
          7586       290.0
          47751      265.0
          ...
          71409      180.0
          105101      72.0
          26632      180.0
          22111      178.0
          21669      161.0
          Name: AQI, Length: 21607, dtype: float64
```

```
In [18]: from sklearn.metrics import mean_absolute_error
          mae = mean_absolute_error(y_test, preds)

          # Print the result
          print(f'Mean Absolute Error (MAE): {mae}')
```

Mean Absolute Error (MAE): 20.125118879050042

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []: