```
import pandas as pd
import numpy as np
from sklearn.linear_model import LinearRegression
import matplotlib.pyplot as plt

import kagglehub


path = kagglehub.dataset_download("uciml/electric-power-consumption-data-set")

print("Path to dataset files:", path)
```

```
Using Colab cache for faster access to the 'electric-power-consumption-data-set' dataset.
Path to dataset files: /kaggle/input/electric-power-consumption-data-set
```

```
import pandas as pd
import os

data_dir = path
file_name = 'household_power_consumption.txt'
file_path = os.path.join(data_dir, file_name)


saldf = pd.read_csv(file_path, sep=';')
saldf.head()
```

```
/tmp/ipython-input-304957410.py:10: DtypeWarning: Columns (2,3,4,5,6,7) have mixed types. Specify dtype option on import or set low_memory=False.
  saldf = pd.read_csv(file_path, sep=';')
```


```
{"type":"dataframe","variable_name":"saldf"}
```


```
saldf.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2075259 entries, 0 to 2075258
Data columns (total 9 columns):
 #   Column                 Dtype 
---  ------                 ----- 
 0   Date                   object
 1   Time                   object
 2   Global_active_power    object
 3   Global_reactive_power  object
 4   Voltage                object
 5   Global_intensity       object
 6   Sub_metering_1         object
 7   Sub_metering_2         object
```

```
 8   Sub_metering_3         float64
dtypes: float64(1), object(8)
memory usage: 142.5+ MB
```

```
saldf.isnull().sum()
```

```
Date                         0
Time                         0
Global_active_power      25979
Global_reactive_power        0
Voltage                      0
Global_intensity         25979
Sub_metering_1               0
Sub_metering_2               0
Sub_metering_3           25979
dtype: int64
```

```
saldf['Global_active_power'] =
pd.to_numeric(saldf['Global_active_power'], errors='coerce')


inp = saldf[['Global_active_power']]
out = saldf['Sub_metering_3']

from sklearn.linear_model import LinearRegression
LR = LinearRegression()


df_combined = pd.concat([inp, out], axis=1)

df_valid = df_combined.dropna()


inp_valid = df_valid[['Global_active_power']]
out_valid = df_valid['Sub_metering_3']


LR.fit(inp_valid, out_valid)
```

```
LinearRegression()
```

```
prediction = LR.predict([[5]])
print("Prediction for Global_active_power = 5:", prediction)
```

```
Prediction for Global_active_power = 5: [26.37417]
```

```
/usr/local/lib/python3.12/dist-packages/sklearn/utils/
validation.py:2739: UserWarning: X does not have valid feature names,
but LinearRegression was fitted with feature names
  warnings.warn(
```