Lab 5.1

Features_extract

Objective:

The objective of this lab is to enhance raw datasets through feature engineering by including calendric (date/time-based) and peak hour information. These features help models better capture time-based patterns in data, such as trends based on days of the week or times of the day.

Introduction:

Feature engineering is a critical step in the data preprocessing pipeline. It involves creating new input features from existing data to improve the performance of machine learning models. Calendric features (such as day of the week, month, or holiday indicator) and peak hour indicators (such as rush hour or working hours) are especially valuable when working with time-series data or datasets with timestamps, such as traffic flow, sales, energy consumption, or customer behavior.

```
import pandas as pd
import numpy as np

df=pd.read_csv(r'C:\Users\PMLS\ML\
LAB5\4_AEP_Introducing_holidays.csv',  parse_dates=['Datetime'])
df.head()
```

```
             Datetime   AEP_MW        Date  Holiday
0 2004-10-01 01:00:00  12379.0  2004-10-01        0
1 2004-10-01 02:00:00  11935.0  2004-10-01        0
2 2004-10-01 03:00:00  11692.0  2004-10-01        0
3 2004-10-01 04:00:00  11597.0  2004-10-01        0
4 2004-10-01 05:00:00  11681.0  2004-10-01        0
```

```
df['Hour']=df['Datetime'].dt.hour
df['Month']=df['Datetime'].dt.month
df['Day_Of_Week']=df['Datetime'].dt.weekday
df["Week"] = df["Datetime"].dt.isocalendar().week
df['yearday'] = df['Datetime'].dt.dayofyear
df['quarter'] = df['Datetime'].dt.quarter
df['weekend']=(df['Day_Of_Week']>=5).astype("int")
df['day_night']=((df['Hour']>=8) & (df['Hour']<=16)).astype("int")
df.head()
```

```
             Datetime   AEP_MW        Date  Holiday  Hour  Month
Day_Of_Week  \
```

```
0 2004-10-01 01:00:00  12379.0  2004-10-01        0     1     10
4
1 2004-10-01 02:00:00  11935.0  2004-10-01        0     2     10
4
2 2004-10-01 03:00:00  11692.0  2004-10-01        0     3     10
4
3 2004-10-01 04:00:00  11597.0  2004-10-01        0     4     10
4
4 2004-10-01 05:00:00  11681.0  2004-10-01        0     5     10
4
```

	Week	yearday	quarter	weekend	day_night
0	40	275	4	0	0
1	40	275	4	0	0
2	40	275	4	0	0
3	40	275	4	0	0
4	40	275	4	0	0

Adding seasons

with Refs. [32–35]. In Panel B of Table 3, LMPs are classified into four sections: 1 = winter (December, January and February), 2 = spring (March, April and May), 3 = summer (June, July and August) and 4 = fall (September, October and November). We see that winter has both the highest average (47.93) and *CV* (1.63), implying that winter effects may exist in the electricity market.

```
df['winter']= ( (df['Month'] == 12) | (df['Month'] ==1) | (df['Month']
==2))*1
df['spring']= ( (df['Month'] == 3) | (df['Month'] ==4) | (df['Month']
==5))*1
df['summer']= ( (df['Month'] == 6) | (df['Month'] ==7) | (df['Month']
==8))*1
df['fall']=   ( (df['Month'] == 9) | (df['Month'] ==10) | (df['Month']
==11))*1
df.tail()
```

	Datetime	AEP_MW	Date	Holiday	Hour	Month
121291	2018-08-02 20:00:00	17673.0	2018-08-02	0	20	8
121292	2018-08-02 21:00:00	17303.0	2018-08-02	0	21	8
121293	2018-08-02 22:00:00	17001.0	2018-08-02	0	22	8

```
121294 2018-08-02 23:00:00  15964.0  2018-08-02        0    23      8

121295 2018-08-03 00:00:00  14809.0  2018-08-03        0     0      8


        Day_Of_Week  Week  yearday  quarter  weekend  day_night
winter  \
121291            3    31      214        3        0          0
0
121292            3    31      214        3        0          0
0
121293            3    31      214        3        0          0
0
121294            3    31      214        3        0          0
0
121295            4    31      215        3        0          0
0

        spring  summer  fall
121291       0       1     0
121292       0       1     0
121293       0       1     0
121294       0       1     0
121295       0       1     0
```

```
df.rename(columns={'AEP_MW': 'aep', 'yearday':
'year_day','Week':'week_no',

'Holiday':'holiday','quarter':'quarter','day_night':'day_night','weeke
nd':'weekend',

'Hour':'hour','Month':'month','Day_Of_Week':'day_of_week'},
inplace=True)
df.head()
```

```
             Datetime      aep        Date  holiday  hour  month
day_of_week  \
0 2004-10-01 01:00:00  12379.0  2004-10-01        0     1     10
4
1 2004-10-01 02:00:00  11935.0  2004-10-01        0     2     10
4
2 2004-10-01 03:00:00  11692.0  2004-10-01        0     3     10
4
3 2004-10-01 04:00:00  11597.0  2004-10-01        0     4     10
4
4 2004-10-01 05:00:00  11681.0  2004-10-01        0     5     10
4

   week_no  year_day  quarter  weekend  day_night  winter  spring
summer  \
```

```
0       40       275        4        0           0       0       0
0
1       40       275        4        0           0       0       0
0
2       40       275        4        0           0       0       0
0
3       40       275        4        0           0       0       0
0
4       40       275        4        0           0       0       0
0
```

```
   fall
0     1
1     1
2     1
3     1
4     1
```

```
df = df.reindex(columns=['Datetime','aep',
'year_day','holiday','weekend','winter','spring','summer','fall','hour
','month','day_of_week'])
df.head()
```

```
             Datetime      aep  year_day  holiday  weekend  winter
spring  \
0 2004-10-01 01:00:00  12379.0       275        0        0       0
0
1 2004-10-01 02:00:00  11935.0       275        0        0       0
0
2 2004-10-01 03:00:00  11692.0       275        0        0       0
0
3 2004-10-01 04:00:00  11597.0       275        0        0       0
0
4 2004-10-01 05:00:00  11681.0       275        0        0       0
0
```

```
   summer  fall  hour  month  day_of_week
0       0     1     1     10            4
1       0     1     2     10            4
2       0     1     3     10            4
3       0     1     4     10            4
4       0     1     5     10            4
```

```
df.to_csv(r'C:\Users\PMLS\ML\LAB5\5_features_extracted.csv',
index=False)
```

```
df.isnull().sum()
```

```
Datetime       0
aep            0
year_day       0
```

```
holiday         0
weekend         0
winter          0
spring          0
summer          0
fall            0
hour            0
month           0
day_of_week     0
dtype: int64
```