

```
In [51]: # This Python 3 environment comes with many helpful analytics libraries installed
# It is defined by the kaggle/python Docker image: https://github.com/kaggle
# For example, here's several helpful packages to load

import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)

# Input data files are available in the read-only "../input/" directory
# For example, running this (by clicking run or pressing Shift+Enter) will list all files in the input directory

import os
for dirname, _, filenames in os.walk('/kaggle/input'):
    for filename in filenames:
        print(os.path.join(dirname, filename))

# You can write up to 20GB to the current directory (/kaggle/working/) that
# You can also write temporary files to /kaggle/temp/, but they won't be saved
```

```
In [52]: df_power=pd.read_csv("https://raw.githubusercontent.com/jenfly/opsd/master/c
```

```
In [53]: df_power.head(10)
```

```
Out[53]:
```

	Date	Consumption	Wind	Solar	Wind+Solar
0	2006-01-01	1069.184	NaN	NaN	NaN
1	2006-01-02	1380.521	NaN	NaN	NaN
2	2006-01-03	1442.533	NaN	NaN	NaN
3	2006-01-04	1457.217	NaN	NaN	NaN
4	2006-01-05	1477.131	NaN	NaN	NaN
5	2006-01-06	1403.427	NaN	NaN	NaN
6	2006-01-07	1300.287	NaN	NaN	NaN
7	2006-01-08	1207.985	NaN	NaN	NaN
8	2006-01-09	1529.323	NaN	NaN	NaN
9	2006-01-10	1576.911	NaN	NaN	NaN

```
In [54]: df_power.dtypes
```

```
Out[54]: Date          object
Consumption    float64
Wind           float64
Solar          float64
Wind+Solar     float64
dtype: object
```

```
In [55]: df_power.describe()
```

```
Out[55]:
```

	Consumption	Wind	Solar	Wind+Solar
<b>count</b>	4383.000000	2920.000000	2188.000000	2187.000000
<b>mean</b>	1338.675836	164.814173	89.258695	272.663481
<b>std</b>	165.775710	143.692732	58.550099	146.319884
<b>min</b>	842.395000	5.757000	1.968000	21.478000
<b>25%</b>	1217.859000	62.353250	35.179250	172.185500
<b>50%</b>	1367.123000	119.098000	86.407000	240.991000
<b>75%</b>	1457.761000	217.900250	135.071500	338.988000
<b>max</b>	1709.568000	826.278000	241.580000	851.556000

```
In [56]: df_power.columns
```

```
Out[56]: Index(['Date', 'Consumption', 'Wind', 'Solar', 'Wind+Solar'], dtype='object')
```

```
In [57]: df_power.shape
```

```
Out[57]: (4383, 5)
```

```
In [58]: print(df_power["Date"])
```

```
0      2006-01-01
1      2006-01-02
2      2006-01-03
3      2006-01-04
4      2006-01-05
...
4378   2017-12-27
4379   2017-12-28
4380   2017-12-29
4381   2017-12-30
4382   2017-12-31
Name: Date, Length: 4383, dtype: object
```

```
In [59]: #convert object to datetime format
df_power['Date'] = pd.to_datetime(df_power['Date'])
```

```
In [60]: print(df_power["Date"])
```

```

0      2006-01-01
1      2006-01-02
2      2006-01-03
3      2006-01-04
4      2006-01-05
...
4378   2017-12-27
4379   2017-12-28
4380   2017-12-29
4381   2017-12-30
4382   2017-12-31
Name: Date, Length: 4383, dtype: datetime64[ns]

```

```
In [61]: df_power = df_power.set_index('Date')
df_power.tail(3)
```

```
Out[61]:
```

	Consumption	Wind	Solar	Wind+Solar
Date				
2017-12-29	1295.08753	584.277	29.854	614.131
2017-12-30	1215.44897	721.247	7.467	728.714
2017-12-31	1107.11488	721.176	19.980	741.156

```
In [62]: df_power.index
```

```
Out[62]: DatetimeIndex(['2006-01-01', '2006-01-02', '2006-01-03', '2006-01-04',
                        '2006-01-05', '2006-01-06', '2006-01-07', '2006-01-08',
                        '2006-01-09', '2006-01-10',
                        ...,
                        '2017-12-22', '2017-12-23', '2017-12-24', '2017-12-25',
                        '2017-12-26', '2017-12-27', '2017-12-28', '2017-12-29',
                        '2017-12-30', '2017-12-31'],
                        dtype='datetime64[ns]', name='Date', length=4383, freq=None)
```

```
In [63]: # Add columns with year, month, and weekday name
df_power['Year'] = df_power.index.year
df_power['Month'] = df_power.index.month
```

```
In [64]: # Display a random sampling of 5 rows
df_power.sample(5)
```

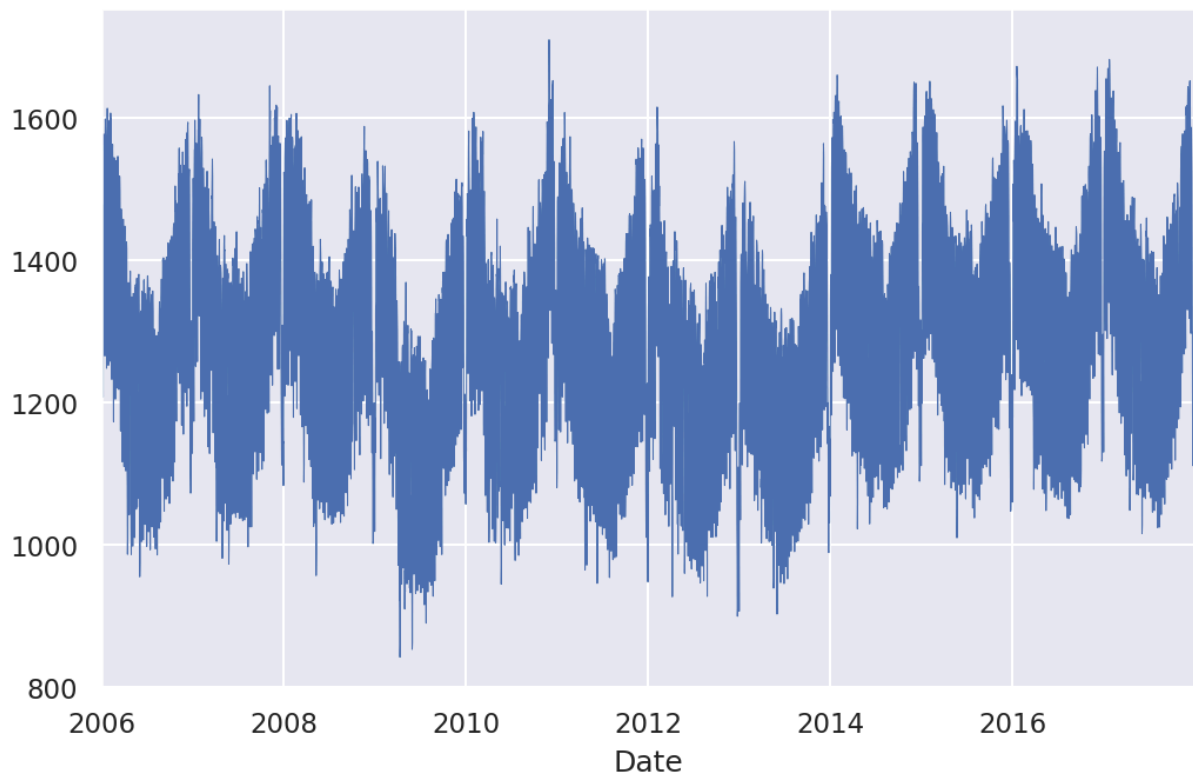
```
Out[64]:
```

	Consumption	Wind	Solar	Wind+Solar	Year	Month
Date						
2006-12-20	1527.593	NaN	NaN	NaN	2006	12
2016-07-15	1362.975	287.953	129.004	416.957	2016	7
2016-02-03	1583.040	534.971	30.859	565.830	2016	2
2017-05-30	1450.411	212.497	166.431	378.928	2017	5
2010-10-29	1434.233	136.641	NaN	NaN	2010	10

```
In [65]: df_power.loc['2015-10-02']
```

```
Out[65]: Consumption    1391.050  
Wind                81.229  
Solar               160.641  
Wind+Solar          241.870  
Year                2015.000  
Month               10.000  
Name: 2015-10-02 00:00:00, dtype: float64
```

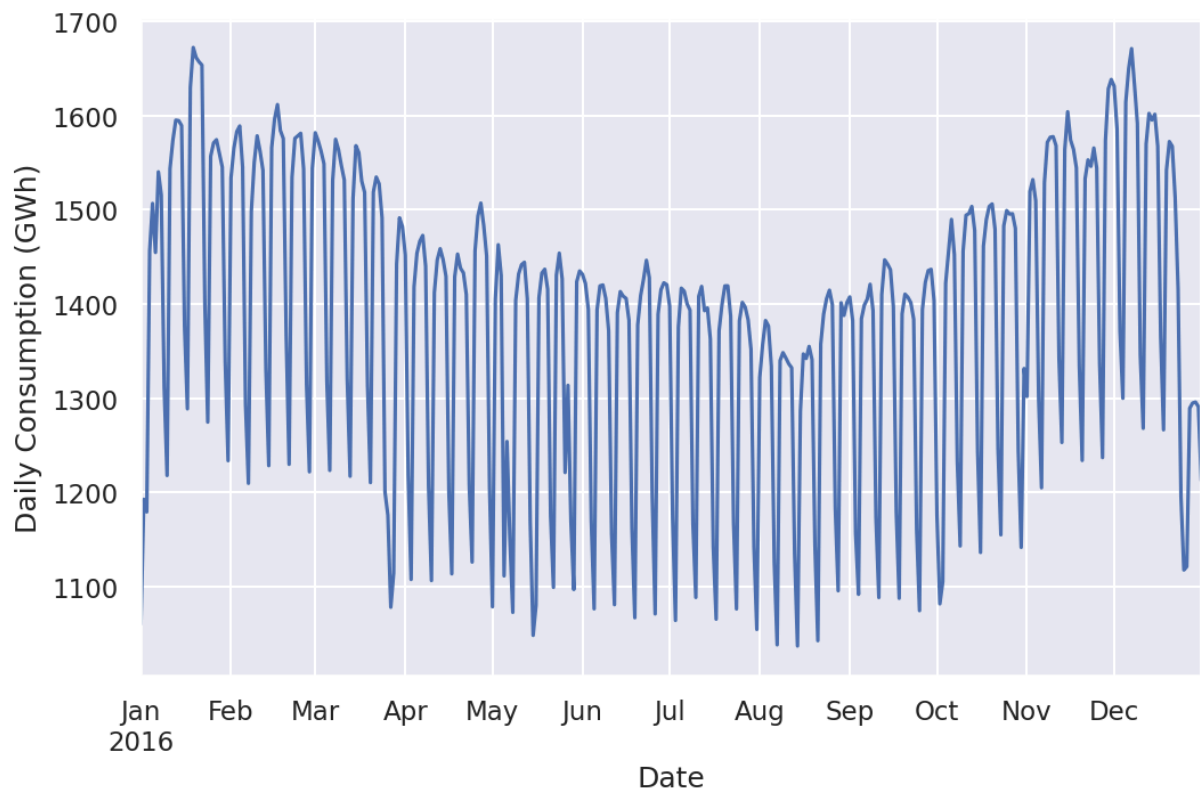
```
In [68]: import matplotlib.pyplot as plt  
import seaborn as sns  
sns.set(rc={'figure.figsize':(11, 4)})  
plt.rcParams['figure.figsize'] = (8,5)  
plt.rcParams['figure.dpi'] = 150  
df_power['Consumption'].plot(linewidth=0.5)  
plt.show()
```



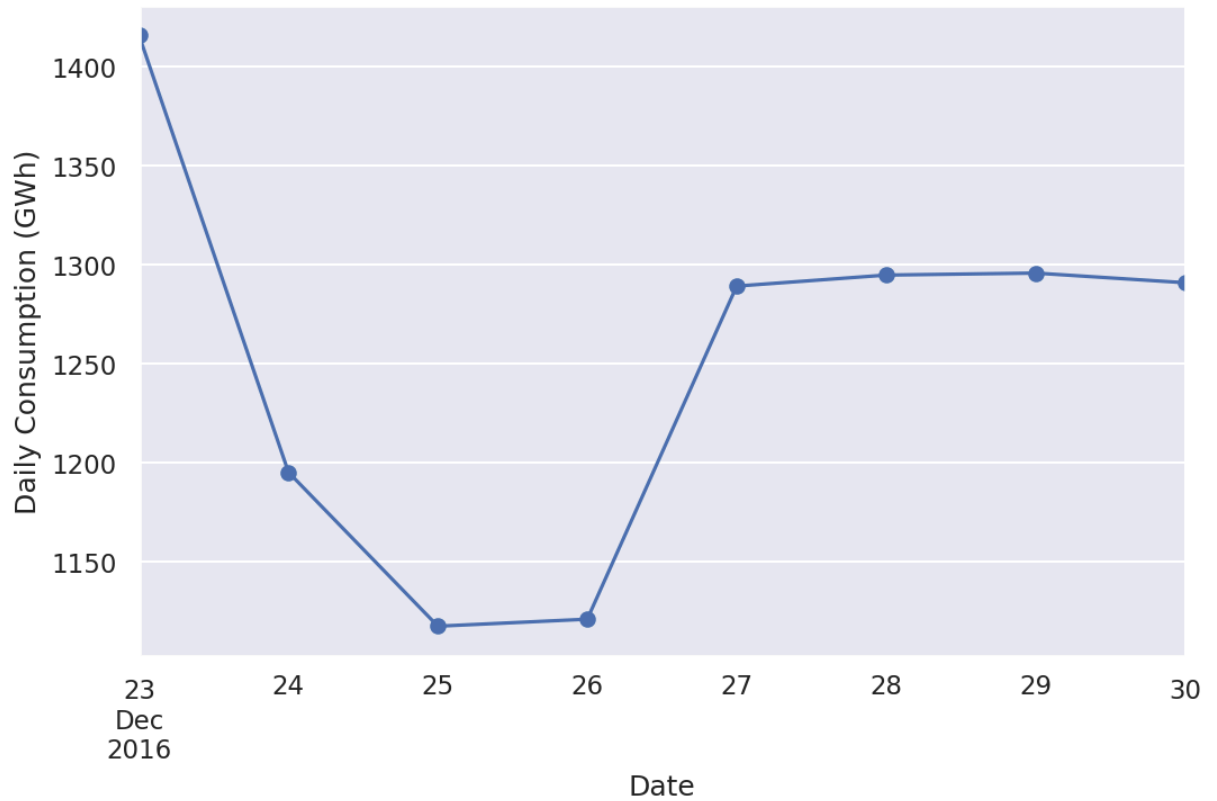
```
In [76]: axes = df_power[['Consumption', 'Solar', 'Wind']].plot(marker='.', alpha=0.5,  
linestyle='None',figsize=(14, 6), subplots=True)
```



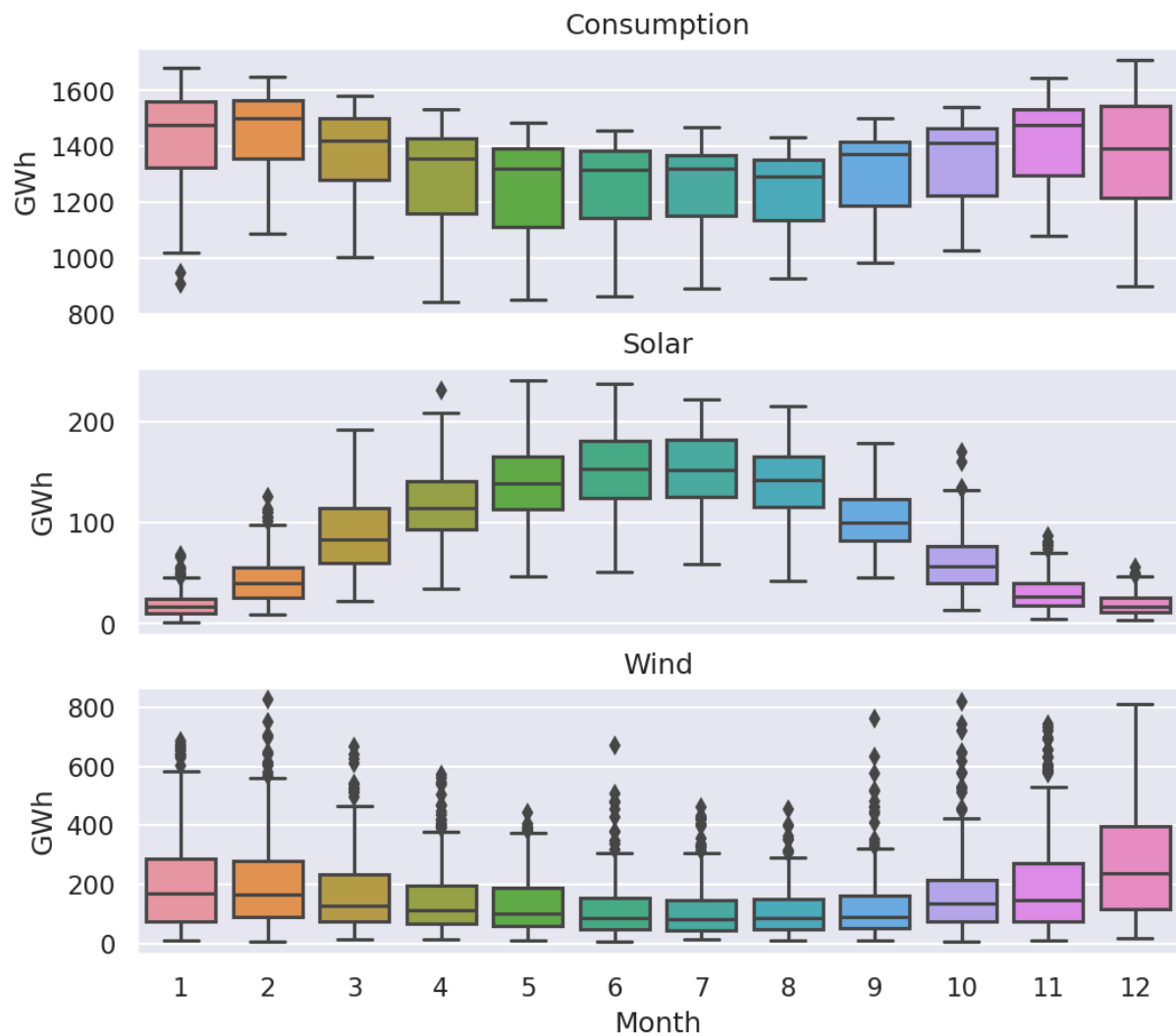
```
In [72]: ax = df_power.loc['2016', 'Consumption'].plot()
ax.set_ylabel('Daily Consumption (GWh)');
```



```
In [78]: ax = df_power.loc['2016-12-23':'2016-12-30', 'Consumption'].plot(marker='o',
ax.set_ylabel('Daily Consumption (GWh)');
```



```
In [80]: fig, axes = plt.subplots(3, 1, figsize=(8, 7), sharex=True)
for name, ax in zip(['Consumption', 'Solar', 'Wind'], axes):
    sns.boxplot(data=df_power, x='Month', y=name, ax=ax)
    ax.set_ylabel('GWh')
    ax.set_title(name)
    if ax != axes[-1]:
        ax.set_xlabel('')
```



```
In [83]: columns = ['Consumption', 'Wind', 'Solar', 'Wind+Solar']
power_weekly_mean = df_power[columns].resample('W').mean()
power_weekly_mean
```

Out[83]:

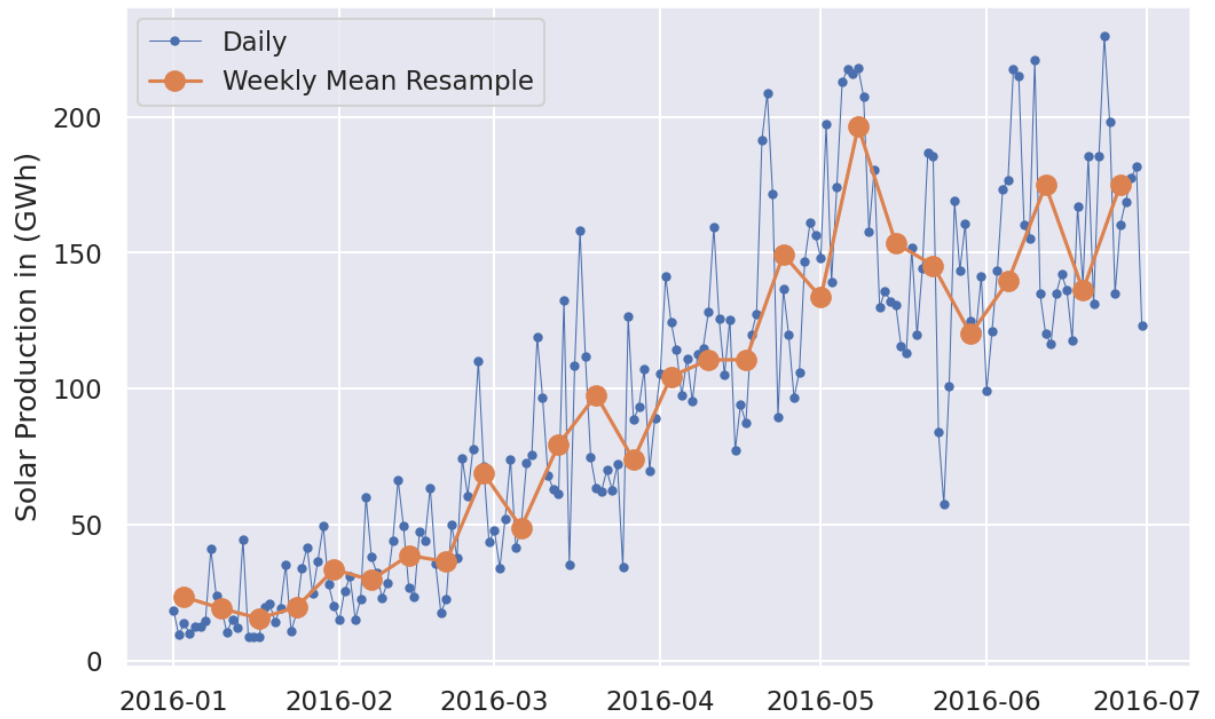
	Consumption	Wind	Solar	Wind+Solar
Date				
2006-01-01	1069.184000	NaN	NaN	NaN
2006-01-08	1381.300143	NaN	NaN	NaN
2006-01-15	1486.730286	NaN	NaN	NaN
2006-01-22	1490.031143	NaN	NaN	NaN
2006-01-29	1514.176857	NaN	NaN	NaN
...	...	...	...	...
2017-12-03	1536.236314	284.334286	18.320857	302.655143
2017-12-10	1554.824946	636.514714	16.440286	652.955000
2017-12-17	1543.856889	442.531857	18.143714	460.675571
2017-12-24	1440.342401	339.018429	9.895143	348.913571
2017-12-31	1203.265211	604.699143	19.240143	623.939286

627 rows × 4 columns

```
In [84]: start, end = '2016-01', '2016-06'
```

```
In [85]: fig, ax = plt.subplots()
ax.plot(df_power.loc[start:end, 'Solar'],
marker='.', linestyle='-', linewidth=0.5, label='Daily')
ax.plot(power_weekly_mean.loc[start:end, 'Solar'],
marker='o', markersize=8, linestyle='-', label='Weekly Mean Resample')
ax.set_ylabel('Solar Production in (GWh)')
ax.legend();
```





In [ ]: