In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
import sklearn
```

In [2]:
```
df=pd.read_csv("../Desktop/sanket/DSBDA/forestfires.csv")
df
```

Out[2]:

	X	Y	month	day	FFMC	DMC	DC	ISI	temp	RH	wind	rain	area
0	7	5	mar	fri	86.2	26.2	94.3	5.1	8.2	51	6.7	0.0	0.00
1	7	4	oct	tue	90.6	35.4	669.1	6.7	18.0	33	0.9	0.0	0.00
2	7	4	oct	sat	90.6	43.7	686.9	6.7	14.6	33	1.3	0.0	0.00
3	8	6	mar	fri	91.7	33.3	77.5	9.0	8.3	97	4.0	0.2	0.00
4	8	6	mar	sun	89.3	51.3	102.2	9.6	11.4	99	1.8	0.0	0.00
...	...	...	...	...	...	...	...	...	...	...	...	...	...
512	4	3	aug	sun	81.6	56.7	665.6	1.9	27.8	32	2.7	0.0	6.44
513	2	4	aug	sun	81.6	56.7	665.6	1.9	21.9	71	5.8	0.0	54.29
514	7	4	aug	sun	81.6	56.7	665.6	1.9	21.2	70	6.7	0.0	11.16
515	1	4	aug	sat	94.4	146.0	614.7	11.3	25.6	42	4.0	0.0	0.00
516	6	3	nov	tue	79.5	3.0	106.7	1.1	11.8	31	4.5	0.0	0.00

517 rows × 13 columns

In [3]:
```
df.head()
```

Out[3]:

	X	Y	month	day	FFMC	DMC	DC	ISI	temp	RH	wind	rain	area
0	7	5	mar	fri	86.2	26.2	94.3	5.1	8.2	51	6.7	0.0	0.0
1	7	4	oct	tue	90.6	35.4	669.1	6.7	18.0	33	0.9	0.0	0.0
2	7	4	oct	sat	90.6	43.7	686.9	6.7	14.6	33	1.3	0.0	0.0
3	8	6	mar	fri	91.7	33.3	77.5	9.0	8.3	97	4.0	0.2	0.0
4	8	6	mar	sun	89.3	51.3	102.2	9.6	11.4	99	1.8	0.0	0.0

In [4]:
```
#data cleaning
df.rename(columns={'rain':'rain_'}, inplace=True)
df
```

Out[4]:

	X	Y	month	day	FFMC	DMC	DC	ISI	temp	RH	wind	rain_	area
0	7	5	mar	fri	86.2	26.2	94.3	5.1	8.2	51	6.7	0.0	0.00
1	7	4	oct	tue	90.6	35.4	669.1	6.7	18.0	33	0.9	0.0	0.00
2	7	4	oct	sat	90.6	43.7	686.9	6.7	14.6	33	1.3	0.0	0.00
3	8	6	mar	fri	91.7	33.3	77.5	9.0	8.3	97	4.0	0.2	0.00
4	8	6	mar	sun	89.3	51.3	102.2	9.6	11.4	99	1.8	0.0	0.00
...	...	...	...	...	...	...	...	...	...	...	...	...	...
512	4	3	aug	sun	81.6	56.7	665.6	1.9	27.8	32	2.7	0.0	6.44
513	2	4	aug	sun	81.6	56.7	665.6	1.9	21.9	71	5.8	0.0	54.29
514	7	4	aug	sun	81.6	56.7	665.6	1.9	21.2	70	6.7	0.0	11.16
515	1	4	aug	sat	94.4	146.0	614.7	11.3	25.6	42	4.0	0.0	0.00
516	6	3	nov	tue	79.5	3.0	106.7	1.1	11.8	31	4.5	0.0	0.00

517 rows × 13 columns

In [5]:
```
#data transformation
df["new_Column"] = pd.NaT
df
```

Out[5]:

	X	Y	month	day	FFMC	DMC	DC	ISI	temp	RH	wind	rain_	area	new_Column
0	7	5	mar	fri	86.2	26.2	94.3	5.1	8.2	51	6.7	0.0	0.00	NaT
1	7	4	oct	tue	90.6	35.4	669.1	6.7	18.0	33	0.9	0.0	0.00	NaT
2	7	4	oct	sat	90.6	43.7	686.9	6.7	14.6	33	1.3	0.0	0.00	NaT
3	8	6	mar	fri	91.7	33.3	77.5	9.0	8.3	97	4.0	0.2	0.00	NaT
4	8	6	mar	sun	89.3	51.3	102.2	9.6	11.4	99	1.8	0.0	0.00	NaT
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
512	4	3	aug	sun	81.6	56.7	665.6	1.9	27.8	32	2.7	0.0	6.44	NaT
513	2	4	aug	sun	81.6	56.7	665.6	1.9	21.9	71	5.8	0.0	54.29	NaT
514	7	4	aug	sun	81.6	56.7	665.6	1.9	21.2	70	6.7	0.0	11.16	NaT
515	1	4	aug	sat	94.4	146.0	614.7	11.3	25.6	42	4.0	0.0	0.00	NaT
516	6	3	nov	tue	79.5	3.0	106.7	1.1	11.8	31	4.5	0.0	0.00	NaT

517 rows × 14 columns

In [6]:
```
#error correcting
df["new_Column"] = pd.NaT
df
```

Out[6]:

	X	Y	month	day	FFMC	DMC	DC	ISI	temp	RH	wind	rain_	area	new_Column
0	7	5	mar	fri	86.2	26.2	94.3	5.1	8.2	51	6.7	0.0	0.00	NaT
1	7	4	oct	tue	90.6	35.4	669.1	6.7	18.0	33	0.9	0.0	0.00	NaT
2	7	4	oct	sat	90.6	43.7	686.9	6.7	14.6	33	1.3	0.0	0.00	NaT
3	8	6	mar	fri	91.7	33.3	77.5	9.0	8.3	97	4.0	0.2	0.00	NaT
4	8	6	mar	sun	89.3	51.3	102.2	9.6	11.4	99	1.8	0.0	0.00	NaT
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
512	4	3	aug	sun	81.6	56.7	665.6	1.9	27.8	32	2.7	0.0	6.44	NaT
513	2	4	aug	sun	81.6	56.7	665.6	1.9	21.9	71	5.8	0.0	54.29	NaT
514	7	4	aug	sun	81.6	56.7	665.6	1.9	21.2	70	6.7	0.0	11.16	NaT
515	1	4	aug	sat	94.4	146.0	614.7	11.3	25.6	42	4.0	0.0	0.00	NaT
516	6	3	nov	tue	79.5	3.0	106.7	1.1	11.8	31	4.5	0.0	0.00	NaT

517 rows × 14 columns

In [7]:
```
df.isnull().sum()
```

Out[7]:
```
X              0
Y              0
month          0
day            0
FFMC           0
DMC            0
DC             0
ISI            0
temp           0
RH             0
wind           0
rain_          0
area           0
new_Column    517
dtype: int64
```

In [8]:
```
df['new_Column'] = df['new_Column'].replace(np.nan, 0)
df.isna().sum()
```

Out[8]:
```
X             0
Y             0
month         0
day           0
FFMC          0
DMC           0
DC            0
ISI           0
temp          0
RH            0
wind          0
rain_         0
area          0
new_Column    0
dtype: int64
```

In [9]:
```
#model building
from sklearn.linear_model import LinearRegression
```

In [10]:
```
A = df['area']
B = df['wind']
```

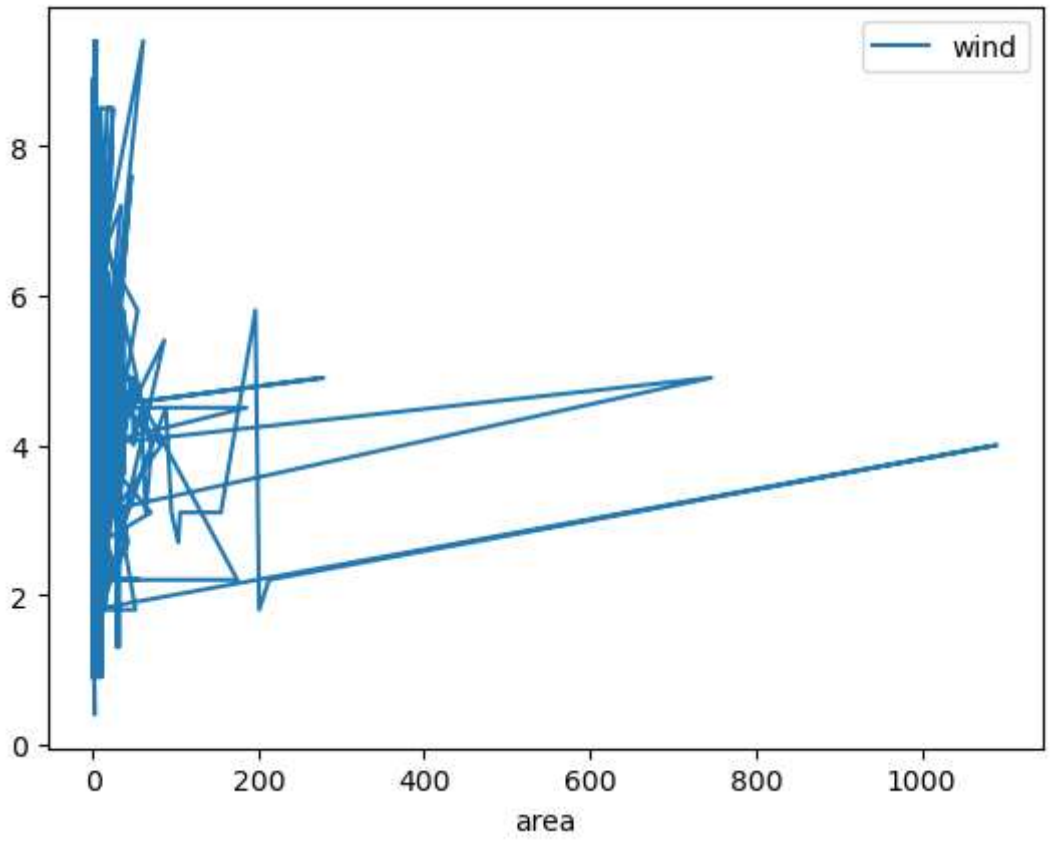
In [11]:
```
lm = LinearRegression(fit_intercept=False)
```

In [12]:
```
lm.fit(df[['area']],df.wind)
```

Out[12]:
```
LinearRegression
LinearRegression(fit_intercept=False)
```

In [13]:
```
df.plot(kind= 'line', x='area' , y='wind')
```

Out[13]:
```
<Axes: xlabel='area'>
```



In [14]:
```
sns.set_theme(style="whitegrid")
df.shape
Q1 = df.quantile(0.25) #first 25% of the data
Q3 = df.quantile(0.75) #first 75% of the data
IQR = Q3 - Q1 #IQR = InterQuartile Range
scale = 2 #For Normal Distributions, scale = 1.5
lower_lim = Q1 - scale*IQR
upper_lim = Q3 + scale*IQR
lower_outliers = (df[df.columns[2:13]] < lower_lim)
upper_outliers = (df[df.columns[2:13]] > upper_lim)
```

```
  Cell In[14], line 10
    upper_outliers = (df[df.columns[2:13]] > upper_lim)
                                                       ^
SyntaxError: invalid non-printable character U+00A0
```

In [ ]: