storm-week2

September 24, 2020

[58]:
```
import pandas as pd
from pandas import read_csv
import matplotlib.pyplot as plt
```

[59]:
```
# Import the 2018 Storm Events data
url = ('StormEvents/StormEvents_2018.csv')
```

[60]:
```
data = pd.read_csv(url)
data.head()
```

[60]:

	EpisodeID	Event_ID	State	Year	Month	Event_Type	\
0	125578	753161	NEBRASKA	2018	June	Hail	
1	125578	753160	NEBRASKA	2018	June	Hail	
2	125988	755273	VERMONT	2018	June	Thunderstorm Wind	
3	125988	755929	VERMONT	2018	June	Thunderstorm Wind	
4	125578	753163	NEBRASKA	2018	June	Tornado	

	Begin_Date_Time	Timezone	End_Date_Time	Injuries_Direct	…	\
0	2018-06-06 18:10:00	MST-7	2018-06-06 18:10:00	0	…	
1	2018-06-06 17:41:00	MST-7	2018-06-06 17:41:00	0	…	
2	2018-06-30 23:30:00	EST-5	2018-06-30 23:32:00	0	…	
3	2018-06-30 23:45:00	EST-5	2018-06-30 23:45:00	0	…	
4	2018-06-06 18:24:00	MST-7	2018-06-06 18:24:00	0	…	

	Damage_Property	Property_Cost	Damage_Crops	Crop_Cost	Begin_Lat	\
0	0.00K	0.0	0.00K	0.0	41.9300	
1	0.00K	0.0	0.00K	0.0	42.0300	
2	15.00K	15000.0	0.00K	0.0	44.9565	
3	10.00K	10000.0	0.00K	0.0	44.7316	
4	0.00K	0.0	0.00K	0.0	40.9000	

	Begin_Lon	End_Lat	End_Lon	\
0	-102.2100	41.9300	-102.2100	
1	-102.1000	42.0300	-102.1000	
2	-72.8699	44.9565	-72.8699	
3	-72.7474	44.7316	-72.7474	
4	-101.7900	40.9000	-101.7900	

```
                                    Episode_Narrative  \
0  Severe storms developed in the Nebraska Panhan…
1  Severe storms developed in the Nebraska Panhan…
2  Vermont and northern NY influenced by heat rid…
3  Vermont and northern NY influenced by heat rid…
4  Severe storms developed in the Nebraska Panhan…

                                     Event_Narrative
0  Hail predominately penny size with some quarte…
1  Hail mainly quarter size with some half dollar…
2       Numerous trees downed by thunderstorm winds.
3  At least half dozen trees downed or snapped al…
4  Tornado briefly touched down in a field 5 mile…

[5 rows x 23 columns]
```

[61]:
```
data.shape
```

[61]:
```
(61742, 23)
```

[62]:
```
data['State'].value_counts()
```

[62]:
```
TEXAS             3370
VIRGINIA          2999
IOWA              2715
KANSAS            2458
SOUTH DAKOTA      2089
                  …
VIRGIN ISLANDS       4
AMERICAN SAMOA       3
HAWAII WATERS        2
GUAM                 1
ST LAWRENCE R        1
Name: State, Length: 67, dtype: int64
```

[63]:
```
# which months have the most tornadoes.
data_month = data.loc[(data['Event_Type'] == 'Tornado')]
print(data_month.shape)
data_month['Month'].value_counts()
```

```
(1248, 23)
```

[63]:
```
May          184
June         159
April        150
October      134
```

```
September    121
November     108
July          97
August        90
December      68
March         66
February      55
January       16
Name: Month, dtype: int64
```

[64]:
```
data['Event_Type'].value_counts()
```

[64]:
```
Thunderstorm Wind             14585
Hail                           7861
Flood                          4715
Winter Weather                 4478
Flash Flood                    4358
Winter Storm                   3375
High Wind                      2944
Drought                        2410
Heavy Snow                     2220
Marine Thunderstorm Wind       2090
Heavy Rain                     1899
Heat                           1282
Tornado                        1248
Strong Wind                    1021
Dense Fog                       752
Blizzard                        748
Frost/Freeze                    701
Extreme Cold/Wind Chill         590
High Surf                       531
Cold/Wind Chill                 523
Excessive Heat                  437
Wildfire                        416
Lightning                       393
Funnel Cloud                    349
Coastal Flood                   320
Tropical Storm                  317
Waterspout                      192
Ice Storm                       171
Debris Flow                     136
Dust Storm                      113
Rip Current                      84
Volcanic Ashfall                 65
Hurricane                        60
Lake-Effect Snow                 56
Dense Smoke                      45
```

```
Marine Tropical Storm            42
Tropical Depression              37
Astronomical Low Tide            33
Storm Surge/Tide                 26
Marine High Wind                 26
Marine Hail                      24
Avalanche                        16
Lakeshore Flood                  10
Marine Strong Wind                9
Sleet                             8
Dust Devil                        8
Marine Hurricane/Typhoon          6
Marine Tropical Depression        5
Freezing Fog                      3
Sneakerwave                       2
Seiche                            2
Name: Event_Type, dtype: int64
```

[65]:
```
# which is the third most frequent event type?
X = data['Event_Type'].value_counts()
X = X.index.tolist()
print(X[2])
```

```
Flood
```

[66]:
```
url = ('hw1.csv')
```

[67]:
```
data2 = pd.read_csv(url)
data2.head()
```

[67]:

	Unnamed: 0	region	area	palmitic	palmitoleic	"stearic"	oleic	\
0	1.North-Apulia	1	1	1075	"75"	"226"	"7823"	
1	2.North-Apulia	1	1	1088	"73"	"224"	"7709"	
2	3.North-Apulia	1	1	911	"54"	"246"	"8113"	
3	4.North-Apulia	1	1	966	"57"	"240"	"7952"	
4	5.North-Apulia	1	1	1051	"67"	"259"	"7771"	

	linoleic	linolenic	arachidic	eicosenoic
0	"672"	"36"	"60"	"29"
1	"781"	"31"	"61"	"29"
2	"549"	"31"	"63"	"29"
3	"619"	"50"	"78"	"35"
4	"672"	"50"	"80"	"46"

[68]:
```
#Identify the data types on each column
data2.dtypes
```

[68]:
```
Unnamed: 0     object
region          int64
area            int64
palmitic        int64
palmitoleic    object
"stearic"      object
oleic          object
linoleic       object
linolenic      object
arachidic      object
eicosenoic     object
dtype: object
```

[69]:
```
data3 = data2.rename(columns={'"stearic"': 'stearic'})

#Delete quotes
data3['palmitoleic'] = data3['palmitoleic'].str.replace(r"[\"\',]", '')
data3['stearic'] = data3['stearic'].str.replace(r"[\"\',]", '')
data3['oleic'] = data3['oleic'].str.replace(r"[\"\',]", '')
data3['linoleic'] = data3['linoleic'].str.replace(r"[\"\',]", '')
data3['linolenic'] = data3['linolenic'].str.replace(r"[\"\',]", '')
data3['arachidic'] = data3['arachidic'].str.replace(r"[\"\',]", '')
data3['eicosenoic'] = data3['eicosenoic'].str.replace(r"[\"\',]", '')

#Change to Int
data3["palmitoleic"] = data3["palmitoleic"].astype(str).astype(int)
data3["stearic"] = data3["stearic"].astype(str).astype(int)
data3["oleic"] = data3["oleic"].astype(str).astype(int)
data3["linoleic"] = data3["linoleic"].astype(str).astype(int)
data3["linolenic"] = data3["linolenic"].astype(str).astype(int)
data3["arachidic"] = data3["arachidic"].astype(str).astype(int)
data3["eicosenoic"] = data3["eicosenoic"].astype(str).astype(int)
```

[70]:
```
data3.head()
```

[70]:

	Unnamed: 0	region	area	palmitic	palmitoleic	stearic	oleic	linoleic	linolenic	arachidic	eicosenoic
0	1.North-Apulia	1	1	1075	75	226	7823	672	36	60	29
1	2.North-Apulia	1	1	1088	73	224	7709	781	31	61	29
2	3.North-Apulia	1	1	911	54	246	8113	549	31	63	29
3	4.North-Apulia	1	1	966	57	240	7952	619	50	78	35
4	5.North-Apulia	1	1	1051	67	259	7771				

```
4        672          50          80          46
```

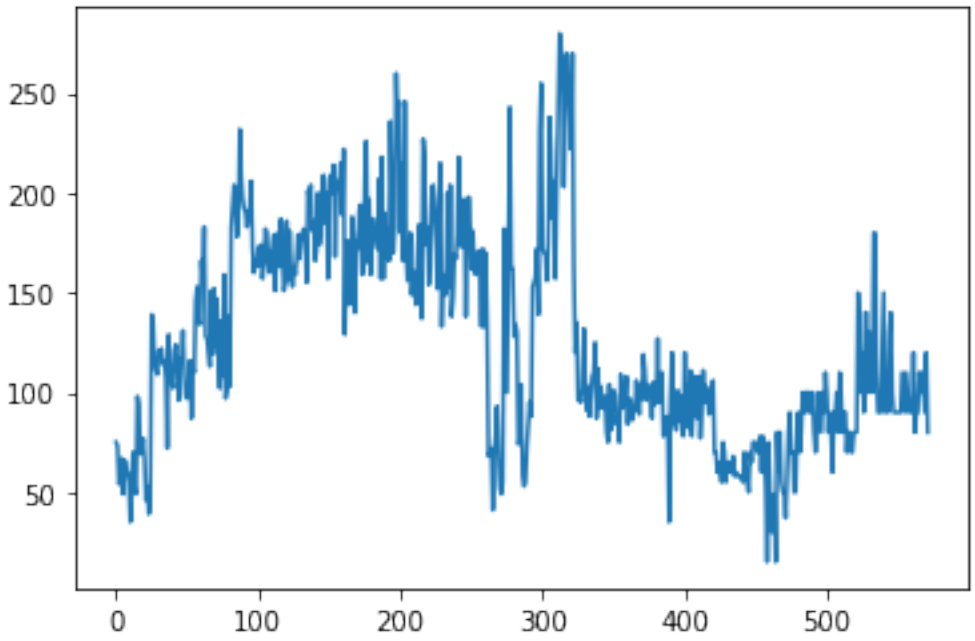
[71]:
```
data3.dtypes
```

[71]:
```
Unnamed: 0     object
region          int64
area            int64
palmitic        int64
palmitoleic     int64
stearic         int64
oleic           int64
linoleic        int64
linolenic       int64
arachidic       int64
eicosenoic      int64
dtype: object
```

[72]:
```
# Plot the values of columns 5 through 11, each column individually, beginning␣
↪from the second row
```

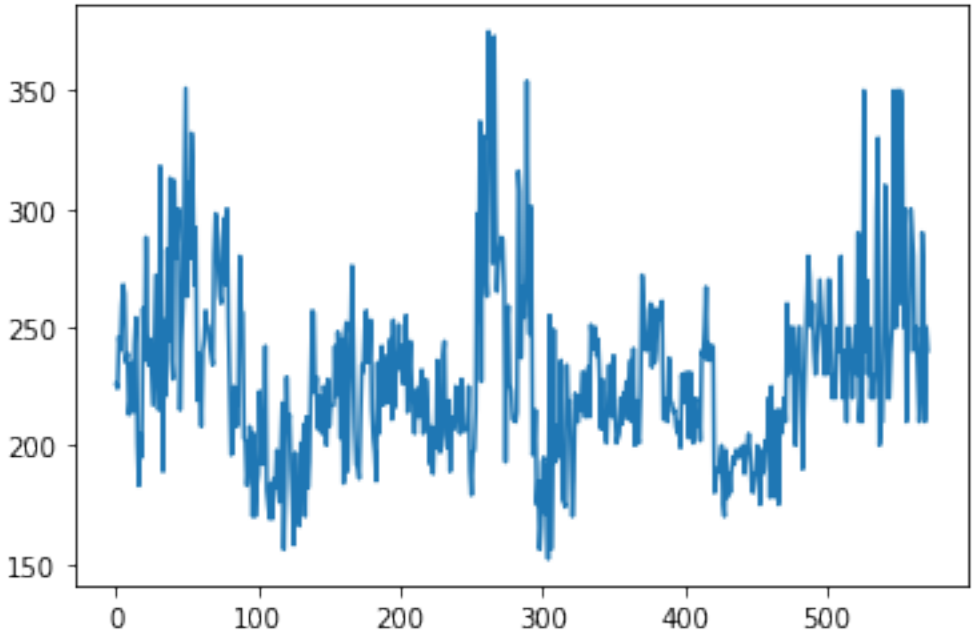
[73]:
```
data3['palmitoleic'].plot()
```

[73]:
```
<AxesSubplot:>
```



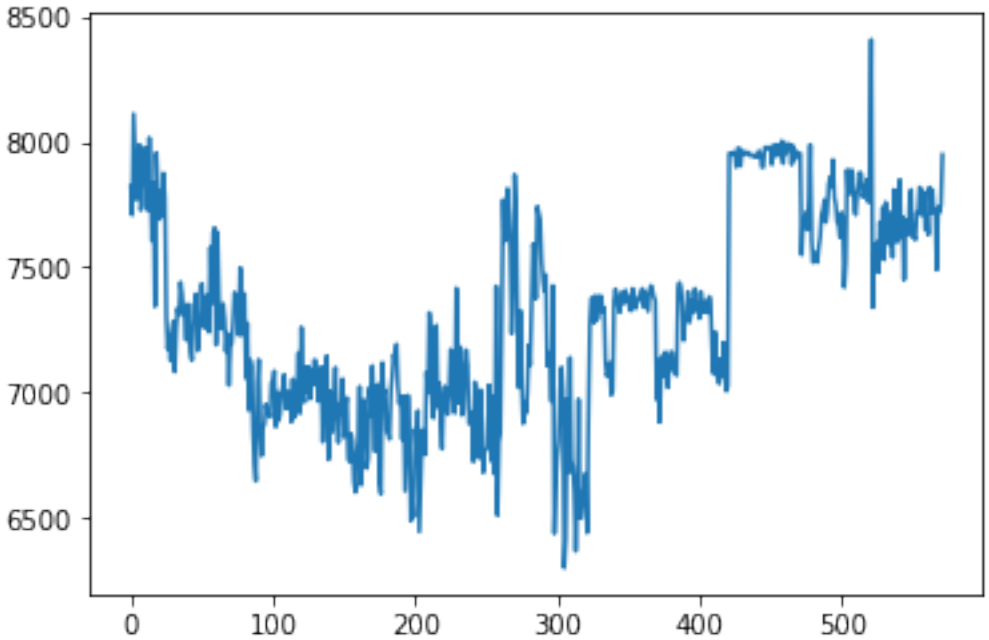
[74]:
```
data3['stearic'].plot()
```

[74]: <AxesSubplot:>



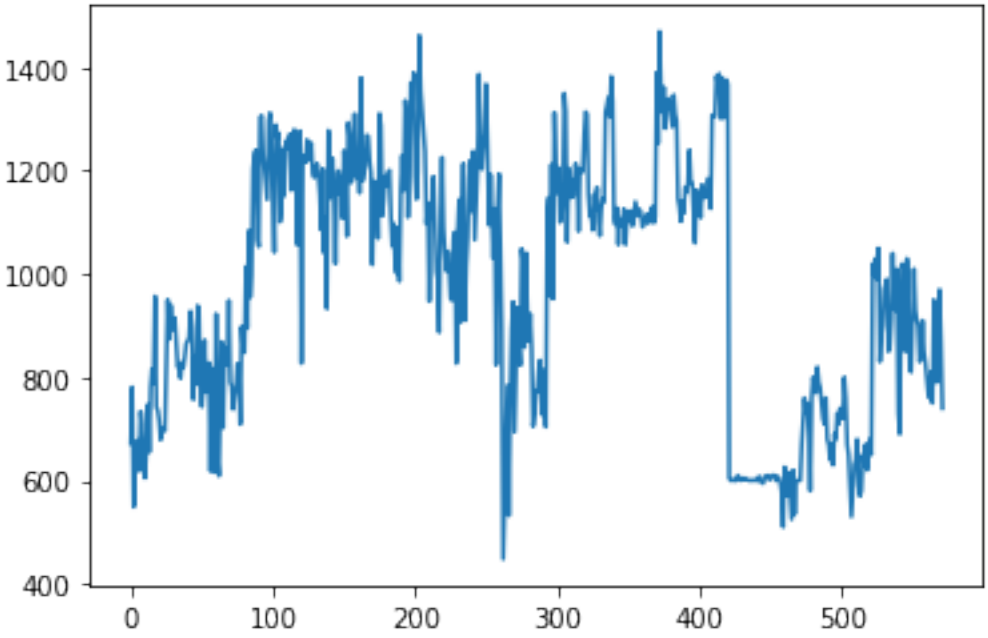
[75]:
```
data3['oleic'].plot()
```

[75]: <AxesSubplot:>



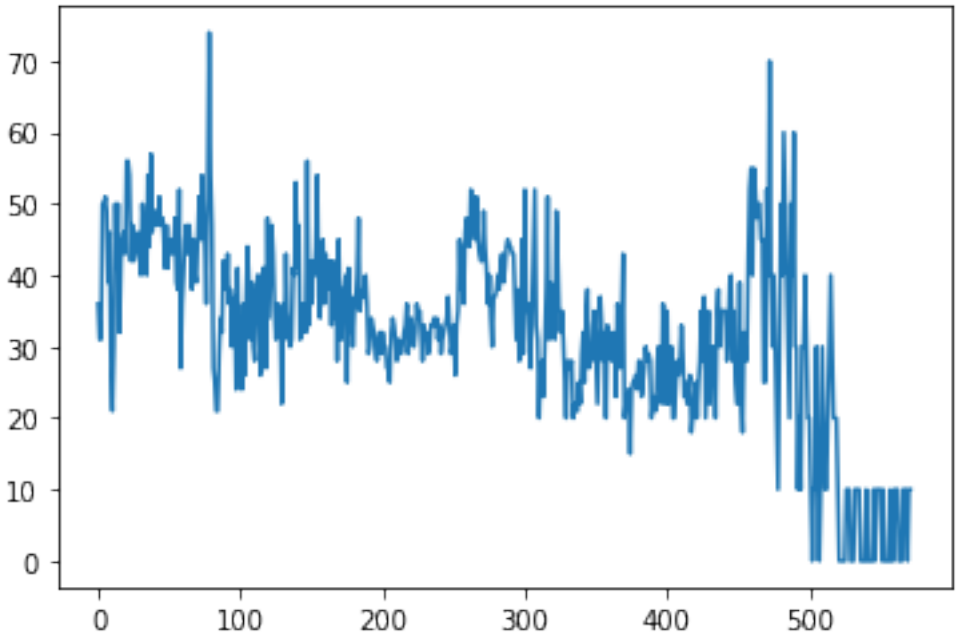
[76]:
```
data3['linoleic'].plot()
```

[76]: <AxesSubplot:>



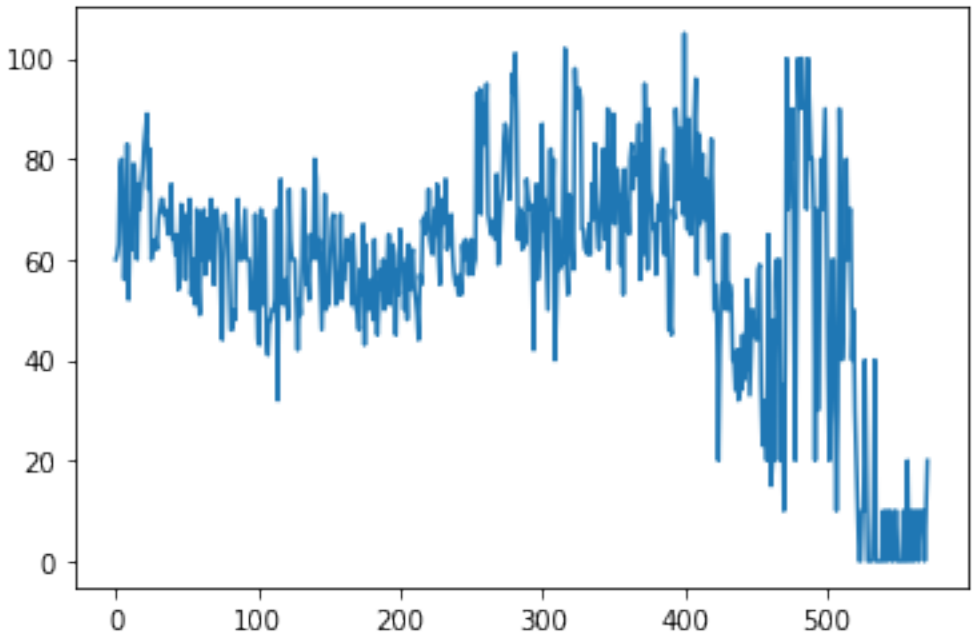
[77]:
```
data3['linolenic'].plot()
```

[77]: <AxesSubplot:>



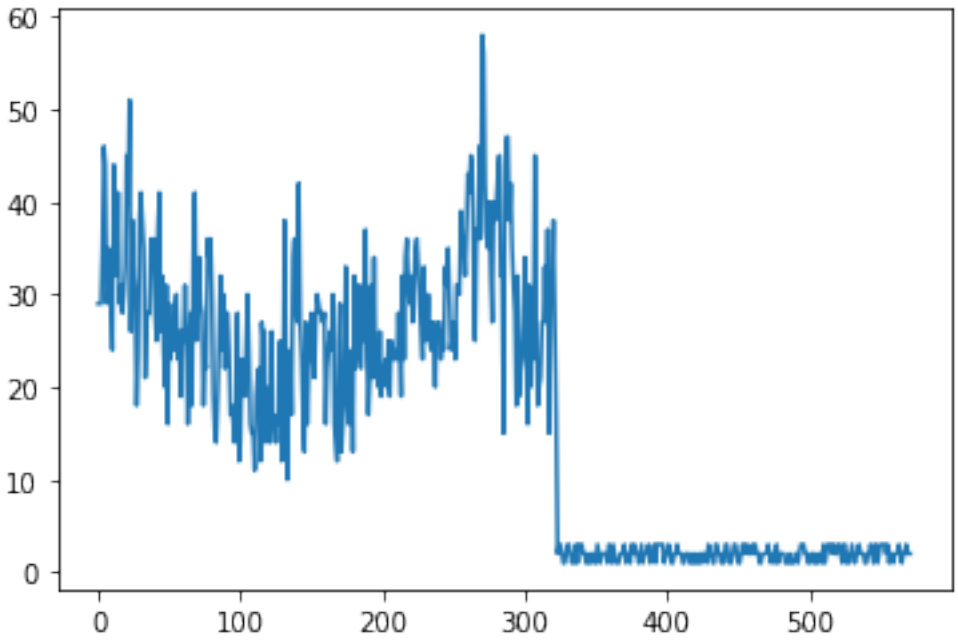
[78]:
```
data3['arachidic'].plot()
```

[78]: <AxesSubplot:>



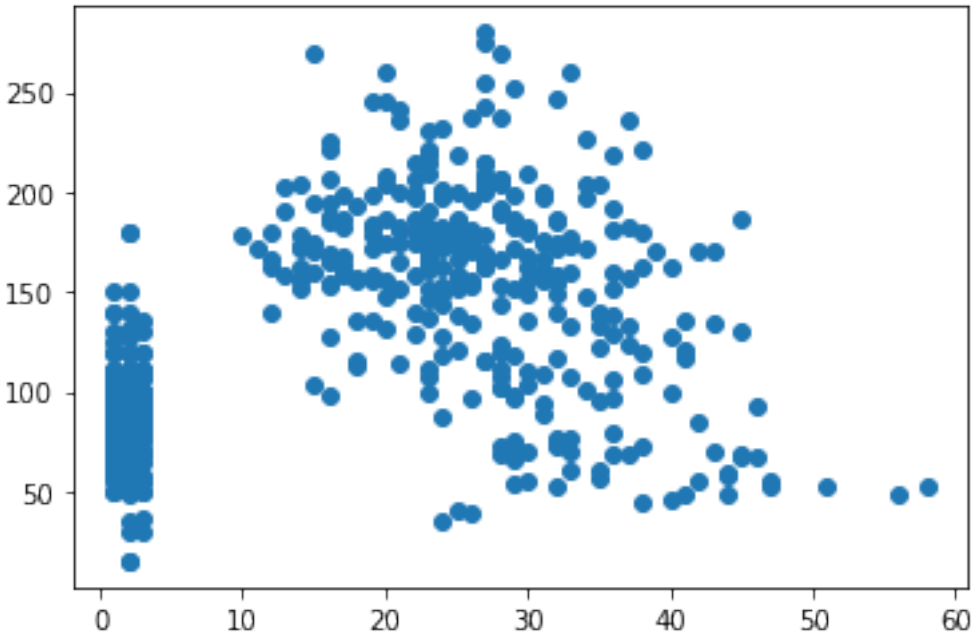
[79]:
```
data3['eicosenoic'].plot()
```

[79]: <AxesSubplot:>



[81]:
```
# Plot the values of columns 11 and 5 together
plt.scatter(data3['eicosenoic'],data3['palmitoleic'])
```

[81]: <matplotlib.collections.PathCollection at 0x11bdaae10>


250
200
150
100
50
0
10
20
30
40
50
60