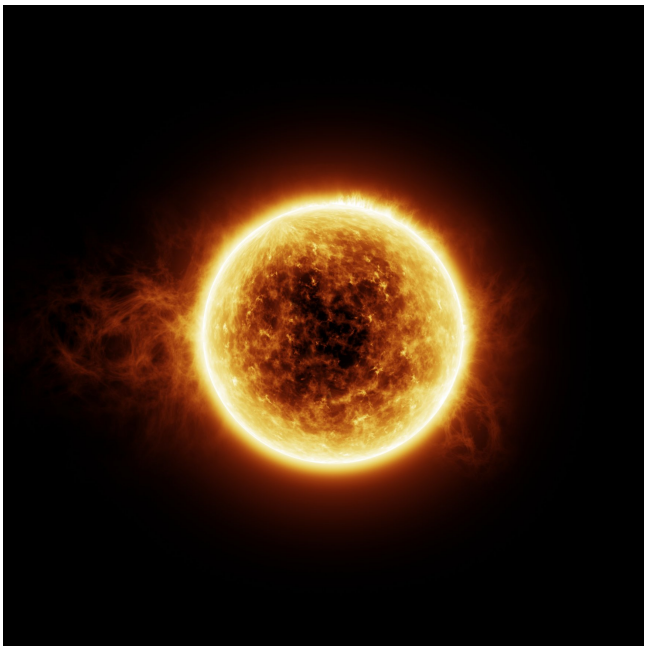
NASA Top 50 Solar Flares



By: Seth Gregory

This project serves to obtain and analyze data regarding the top 50 solar flares recorded by NASA since 1998. It is broken up into several part: first, data about solar flares is scraped from both the NASA site, and another site with information on the top 50 flares, SpaceWeatherLive.com, the latter of which is used to compare with the NASA data. After this, the data is organized, and the top 50 flares from the NASA site are matched with their likely counterparts. In this way the top 50 flares from the NASA site are designated. Finally, the data is used to consider the claim that the biggest solar flares tend to occur in tandem with a cluster of flares.

Scraping the SpaceWeatherLive (SWL) Data:

First, we will obtain the data from SpaceWeatherLive.com using BeautifulSoup to scrape the HTML data.

In [171…

```
import requests
r = requests.get('https://cmsc320.github.io/files/top-50-solar-flares.html')
```

In [172…

```
from bs4 import BeautifulSoup
soup = BeautifulSoup(r.text, 'html.parser')

# Output suppressed, but used to inform the following section
# soup.prettify()
```

In [173…

```
baseData = soup.find('table', { 'class' : 'table table-striped table-responsi
```

In [174…

```
rows = baseData.findChildren('tr')
table = []

for row in rows:
    cells = row.findChildren('td')
    text = []

    for cell in cells:
        text.append(cell.text)

    table.append(text)
```

Here, we use pandas to place the data into a table.

In [175…

```
import pandas as pd
import numpy as np

pd.set_option('display.max_rows', 10)
frame = pd.DataFrame(table, columns=['rank', 'x_class', 'date', 'region', 'st
frame
```

Out[175…

	rank	x_class	date	region	start_time	max_time	end_time	movie
1	1	X28+	2003/11/04	0486	19:29	19:53	20:06	MovieView archive
2	2	X20+	2001/04/02	9393	21:32	21:51	22:03	MovieView archive
3	3	X17.2+	2003/10/28	0486	09:51	11:10	11:24	MovieView archive
4	4	X17+	2005/09/07	0808	17:17	17:40	18:03	MovieView archive
5	5	X14.4	2001/04/15	9415	13:19	13:50	13:55	MovieView archive
…	…	…	…	…	…	…	…	…
46	46	X2.7	2015/05/05	2339	22:05	22:11	22:15	MovieView archive
47	47	X2.7	2003/11/03	0488	01:09	01:30	01:45	MovieView archive
48	48	X2.7	1998/05/06	8210	07:58	08:09	08:20	MovieView archive
49	49	X2.6	2005/01/15	0720	22:25	23:02	23:31	MovieView archive
50	50	X2.6	2001/09/24	9632	09:32	10:38	11:09	MovieView archive

50 rows × 8 columns

Tidying the SWL Data:

Now that we've got the SpaceWeatherLive Data into a table, we can change things around a bit to make it clearer to look at. First, we drop the last column, since it's redundant, then we combine the date and time columns to make start, end, and max datetime columns. The result is a final table of the SWL data that we will reference later.

In [95]:
```
import datetime as dt

updated_frame = frame.drop(columns='movie')

for index, row in updated_frame.iterrows():
    date = row['date'].split('/')
    start_time = row['start_time'].split(':')
    max_time = row['max_time'].split(':')
    end_time = row['end_time'].split(':')

    start_dt = dt.datetime(int(date[0]), int(date[1]), int(date[2]), int(star
    max_dt = dt.datetime(int(date[0]), int(date[1]), int(date[2]), int(max_ti
    end_dt = dt.datetime(int(date[0]), int(date[1]), int(date[2]), int(end_ti

    updated_frame.at[index, 'start_time'] = start_dt
    updated_frame.at[index, 'max_time'] = max_dt
    updated_frame.at[index, 'end_time'] = end_dt

updated_frame = updated_frame.drop(columns='date')
updated_frame = updated_frame.reindex(columns=['rank', 'x_class', 'start_time
updated_frame = updated_frame.rename(columns={'start_time': 'start_datetime',

updated_frame
```

Out[95]:

	rank	x_class	start_datetime	max_datetime	end_datetime	region
1	1	X28+	2003-11-04 19:29:00	2003-11-04 19:53:00	2003-11-04 20:06:00	0486
2	2	X20+	2001-04-02 21:32:00	2001-04-02 21:51:00	2001-04-02 22:03:00	9393
3	3	X17.2+	2003-10-28 09:51:00	2003-10-28 11:10:00	2003-10-28 11:24:00	0486
4	4	X17+	2005-09-07 17:17:00	2005-09-07 17:40:00	2005-09-07 18:03:00	0808
5	5	X14.4	2001-04-15 13:19:00	2001-04-15 13:50:00	2001-04-15 13:55:00	9415
...	...	...	...	...	...	...
46	46	X2.7	2015-05-05 22:05:00	2015-05-05 22:11:00	2015-05-05 22:15:00	2339
47	47	X2.7	2003-11-03 01:09:00	2003-11-03 01:30:00	2003-11-03 01:45:00	0488
48	48	X2.7	1998-05-06 07:58:00	1998-05-06 08:09:00	1998-05-06 08:20:00	8210
49	49	X2.6	2005-01-15 22:25:00	2005-01-15 23:02:00	2005-01-15 23:31:00	0720
50	50	X2.6	2001-09-24 09:32:00	2001-09-24 10:38:00	2001-09-24 11:09:00	9632

50 rows × 6 columns

Scraping the NASA Data:

Now, using BeautifulSoup once again, we'll scrape the data from the NASA site and place it into a table. There's more data for each flare this time, so the table will be several columns larger.

In [176…

```
import requests
from bs4 import BeautifulSoup

nasa = requests.get('https://cdaw.gsfc.nasa.gov/CME_list/radio/waves_type2.ht
nasa_soup = BeautifulSoup(nasa.text, 'html.parser')

# Suppressed
# nasa_soup.prettify()
```

In [180…

```
import pandas as pd
import numpy as np

lines = nasa_soup.find('pre').get_text().splitlines()[12:]
split_lines = []
for line in lines:
    line_string = str(line)
    split = line_string.split()[:14]
    split_lines.append(split)

nasa_frame = pd.DataFrame(split_lines, columns=['start_date', 'start_time', '
nasa_frame = nasa_frame.drop(index=518)
nasa_frame
```

Out[180…

	start_date	start_time	end_date	end_time	start_frequency	end_frequency	flare_location
0	1997/04/01	14:00	04/01	14:15	8000	4000	S25E16
1	1997/04/07	14:30	04/07	17:30	11000	1000	S28E19
2	1997/05/12	05:15	05/14	16:00	12000	80	N21W08
3	1997/05/21	20:20	05/21	22:00	5000	500	N05W12
4	1997/09/23	21:53	09/23	22:16	6000	2000	S29E25
...	...	...	...	...	...	...	...
513	2017/09/04	20:27	09/05	04:54	14000	210	S10W12
514	2017/09/06	12:05	09/07	08:00	16000	70	S08W33
515	2017/09/10	16:02	09/11	06:50	16000	150	S09W92
516	2017/09/12	07:38	09/12	07:43	16000	13000	N08E48
517	2017/09/17	11:45	09/17	12:35	16000	900	S08E170

518 rows × 14 columns

Tidying the NASA Data:

Like before, now we will organize the NASA data a bit better, combining and adding some columns. First, we recode any missing entries as NaN. Next, we create three datetime columns, as before. Then, we handle the case of "Halo" flares in the cme_angle column by creating a new column with "true" or "false" corresponding to whether or not the flare is a Halo flare or not. Finally, we remove the lower bound indicators from the cme_width column, replacing it with a new column that signifies whether the given flare's width is given as a lower bound.

In [178…

```
import datetime as dt

# Signify missing data
updated_nasa_frame = nasa_frame.replace(['????', '-----', '----', '---', '--:

# Function for creating is_halo column
def is_halo(n):
    return n == 'Halo'

# Add is_halo column and replace 'Halo' values with NaN
cme_angles = updated_nasa_frame['cme_angle']
updated_nasa_frame['is_halo'] = list(map(is_halo, cme_angles))
updated_nasa_frame = updated_nasa_frame.replace('Halo', np.nan)

# Function for creating width_lower_bound column
def is_lower_bound(n):
    return '>' in str(n)

# Add width_lower_bound column and remove non-numeric values from cme_width
cme_width = updated_nasa_frame['cme_width']
updated_nasa_frame['width_lower_bound'] = list(map(is_lower_bound, cme_width)
updated_nasa_frame = updated_nasa_frame.replace(to_replace='>', value='', reg

for index, row in updated_nasa_frame.iterrows():
    # Create start datetime column
    start_date = row['start_date'].split('/')
    start_time = row['start_time'].split(':')
    start_dt = dt.datetime(int(start_date[0]), int(start_date[1]), int(start_
    updated_nasa_frame.at[index, 'start_time'] = start_dt

    # Create end datetime column
    end_date = row['end_date'].split('/')
    end_time = row['end_time'].split(':')
    increment_day = False;

    # Check if end time passes midnight
    if end_time[0] == '24':
        end_time[0] = '00'
        increment_day = True;

    end_dt = dt.datetime(int(start_date[0]), int(end_date[0]), int(end_date[1

    if increment_day:
        end_dt += dt.timedelta(days=1)

    updated_nasa_frame.at[index, 'end_time'] = end_dt

    # Create cme datetime column, checking for missing data
    try:
        cme_date = row['cme_date'].split('/')
        cme_time = row['cme_time'].split(':')

        cme_dt = dt.datetime(int(start_date[0]), int(cme_date[0]), int(cme_da
```

```
    except AttributeError as e:
        cme_dt = np.nan

    updated_nasa_frame.at[index, 'cme_time'] = cme_dt

# Update the columns
updated_nasa_frame = updated_nasa_frame.drop(columns=['start_date','end_date'
updated_nasa_frame = updated_nasa_frame.reindex(columns=['start_time', 'end_t
updated_nasa_frame = updated_nasa_frame.rename(columns={'start_time': 'start_

updated_nasa_frame
```

Out[178...

	start_datetime	end_datetime	start_frequency	end_frequency	flare_location	flare_region
1	1997-04-07 14:30:00	1997-04-07 17:30:00	11000	1000	S28E19	8027
2	1997-05-12 05:15:00	1997-05-14 16:00:00	12000	80	N21W08	8038
3	1997-05-21 20:20:00	1997-05-21 22:00:00	5000	500	N05W12	8040
4	1997-09-23 21:53:00	1997-09-23 22:16:00	6000	2000	S29E25	8088
5	1997-11-03 05:15:00	1997-11-03 12:00:00	14000	250	S20W13	8100
...	...	...	...	...	...	...
513	2017-09-04 20:27:00	2017-09-05 04:54:00	14000	210	S10W12	12673
514	2017-09-06 12:05:00	2017-09-07 08:00:00	16000	70	S08W33	12673
515	2017-09-10 16:02:00	2017-09-11 06:50:00	16000	150	S09W92	NaN
516	2017-09-12 07:38:00	2017-09-12 07:43:00	16000	13000	N08E48	12680
517	2017-09-17 11:45:00	2017-09-17 12:35:00	16000	900	S08E170	NaN

517 rows × 13 columns

Replication & Integration of the SolarWeatherLive Data:

Using our NASA data, we would now like to create our own top 50 solar flares list. To do so, we will use the SWL data to find matches in the NASA data referring to the same flares. To begin, since all flares in the SWL list are of the X-class, we can refine our search in the NASA data to just the X-class flares:

In [181…

```
# Filter out all non-xclass flares
removed_missing = updated_nasa_frame[pd.notnull(updated_nasa_frame['flare_cla
nasa_flares_xclass = removed_missing[removed_missing['flare_classification'].

def remove_x(n):
    newstr = n.replace("X", "")
    return float(newstr)

def add_x(n):
    newstr = "X" + str(n)
    return newstr


nasa_flares_xclass['flare_classification'] = nasa_flares_xclass['flare_classi
nasa_flares_xclass = nasa_flares_xclass.sort_values(by='flare_classification'
nasa_flares_xclass['flare_classification'] = nasa_flares_xclass['flare_classi
nasa_flares_xclass = nasa_flares_xclass.iloc[::-1]

nasa_flares_xclass
```

Out[181…

	start_datetime	end_datetime	start_frequency	end_frequency	flare_location	flare_region
240	2003-11-04 20:00:00	2003-11-05 00:00:00	10000	200	S19W83	10486
117	2001-04-02 22:05:00	2001-04-03 02:30:00	14000	250	N19W72	9393
233	2003-10-28 11:10:00	2003-10-30 00:00:00	14000	40	S16E08	10486
126	2001-04-15 14:05:00	2001-04-16 13:00:00	14000	40	S20W85	9415
234	2003-10-29 20:55:00	2003-10-30 00:00:00	11000	500	S15W02	10486
…	…	…	…	…	…	…
80	2000-07-11 13:00:00	2000-07-11 13:30:00	12000	1000	N18E27	9077
428	2013-11-19 10:39:00	2013-11-19 20:20:00	14000	100	S14W70	11893
16	1998-04-27 09:20:00	1998-04-27 10:00:00	10000	1000	S16E50	8210
153	2001-11-04 16:30:00	2001-11-06 11:00:00	14000	70	N06W18	9684
196	2002-08-03 19:20:00	2002-08-03 20:30:00	14000	2000	S16W76	10039

92 rows × 13 columns

Comparing the SWL and NASA data side by side:

Below we compare the data, but notice that some of the NASA results do not appear in the SWL results. Thus, one potential explanation is that their X-class data was missing in the NASA data. So we have to broaden our search.

In [183…

```
# Function for displaying two tables side by side
from IPython.display import display_html
def display_side(*args):
    t=''
    for data_frame in args:
        t+=data_frame.to_html()
    display_html(t.replace('table', 'table style="display:inline"'), raw=True


top_nasa_data = nasa_flares_xclass.copy()
swl_data = updated_frame.copy()

# Removes the non-numeric characters from the x_class strings
def remove_nonnum(n):
    newstr = n.replace("+", "")
    newstr = newstr.replace("X", "")
    return float(newstr)


swl_data['x_class'] = swl_data['x_class'].apply(remove_nonnum)
swl_data = swl_data.drop(columns=['max_datetime', 'end_datetime'])

top_nasa_data = top_nasa_data.drop(columns=['end_datetime', 'start_frequency'
top_nasa_data['flare_classification'] = top_nasa_data['flare_classification']

# Cleans the region data to match the format of the SWL data
def edit_region(n):
    if len(str(n)) > 4:
        return n[1:]
    else:
        return n

top_nasa_data['flare_region'] = list(map(edit_region, top_nasa_data['flare_re
display_side(swl_data, nasa_data)
```

	rank	x_class	start_datetime	region
1	1	28.0	2003-11-04 19:29:00	0486
2	2	20.0	2001-04-02 21:32:00	9393
3	3	17.2	2003-10-28 09:51:00	0486
4	4	17.0	2005-09-07 17:17:00	0808
5	5	14.4	2001-04-15 13:19:00	9415
6	6	10.0	2003-10-29 20:37:00	0486
7	7	9.4	1997-11-06 11:49:00	8100
8	8	9.3	2017-09-06 11:53:00	2673
9	9	9.0	2006-12-05 10:18:00	0930

10	10	8.3	2003-11-02 17:03:00	0486
11	11	8.2	2017-09-10 15:35:00	2673
12	12	7.1	2005-01-20 06:36:00	0720
13	13	6.9	2011-08-09 07:48:00	1263
14	14	6.5	2006-12-06 18:29:00	0930
15	15	6.2	2005-09-09 19:13:00	0808
16	16	6.2	2001-12-13 14:20:00	9733
17	17	5.7	2000-07-14 10:03:00	9077
18	18	5.6	2001-04-06 19:10:00	9415
19	19	5.4	2012-03-07 00:02:00	1429
20	20	5.4	2005-09-08 20:52:00	0808
21	21	5.4	2003-10-23 08:19:00	0486
22	22	5.3	2001-08-25 16:23:00	9591
23	23	4.9	2014-02-25 00:39:00	1990
24	24	4.9	1998-08-18 22:10:00	8307
25	25	4.8	2002-07-23 00:18:00	0039
26	26	4.0	2000-11-26 16:34:00	9236
27	27	3.9	2003-11-03 09:43:00	0488
28	28	3.9	1998-08-19 21:35:00	8307
29	29	3.8	2005-01-17 06:59:00	0720
30	30	3.7	1998-11-22 06:30:00	8384
31	31	3.6	2005-09-09 09:42:00	0808
32	32	3.6	2004-07-16 13:49:00	0649
33	33	3.6	2003-05-28 00:17:00	0365
34	34	3.4	2006-12-13 02:14:00	0930
35	35	3.4	2001-12-28 20:02:00	9767
36	36	3.3	2013-11-05 22:07:00	1890
37	37	3.3	2002-07-20 21:04:00	0039
38	38	3.3	1998-11-28 04:54:00	8395
39	39	3.2	2013-05-14 00:00:00	1748
40	40	3.1	2014-10-24 21:07:00	2192
41	41	3.1	2002-08-24 00:49:00	0069

42	42	3.0	2002-07-15 19:59:00	0030
43	43	2.8	2013-05-13 15:48:00	1748
44	44	2.8	2001-12-11 07:58:00	9733
45	45	2.8	1998-08-18 08:14:00	8307
46	46	2.7	2015-05-05 22:05:00	2339
47	47	2.7	2003-11-03 01:09:00	0488
48	48	2.7	1998-05-06 07:58:00	8210
49	49	2.6	2005-01-15 22:25:00	0720
50	50	2.6	2001-09-24 09:32:00	9632

	start_datetime	flare_region	flare_classification
240	2003-11-04 20:00:00	0486	28.0
117	2001-04-02 22:05:00	9393	20.0
233	2003-10-28 11:10:00	0486	17.0
126	2001-04-15 14:05:00	9415	14.0
234	2003-10-29 20:55:00	0486	10.0
8	1997-11-06 12:20:00	8100	9.4
514	2017-09-06 12:05:00	2673	9.3
328	2006-12-05 10:50:00	0930	9.0
237	2003-11-02 17:30:00	0486	8.3
515	2017-09-10 16:02:00	NaN	8.3
288	2005-01-20 07:15:00	0720	7.1
359	2011-08-09 08:20:00	1263	6.9
331	2006-12-06 19:00:00	0930	6.5
317	2005-09-09 19:45:00	0808	6.2
82	2000-07-14 10:30:00	9077	5.7
121	2001-04-06 19:35:00	9415	5.6
375	2012-03-07 01:00:00	1429	5.4
135	2001-08-25 16:50:00	9591	5.3
443	2014-02-25 00:56:00	1990	4.9
193	2002-07-23 00:50:00	0039	4.8
104	2000-11-26 17:00:00	9236	4.0
239	2003-11-03 10:00:00	0488	3.9

286	2005-01-17 10:00:00	0720	3.8
222	2003-05-28 01:00:00	0365	3.6
160	2001-12-28 20:35:00	9756	3.4
332	2006-12-13 02:45:00	0930	3.4
192	2002-07-20 21:30:00	0039	3.3
404	2013-05-14 01:16:00	1748	3.2
201	2002-08-24 01:45:00	0069	3.1
403	2013-05-13 16:15:00	1748	2.8
19	1998-05-06 08:25:00	8210	2.7
487	2015-05-05 22:24:00	2339	2.7
238	2003-11-03 01:15:00	0488	2.7
142	2001-09-24 10:45:00	9632	2.6
9	1997-11-27 13:30:00	8113	2.6
284	2005-01-15 23:00:00	0720	2.6
276	2004-11-10 02:25:00	0696	2.5
123	2001-04-10 05:24:00	9415	2.3
99	2000-11-24 15:25:00	9236	2.3
73	2000-06-06 15:20:00	9026	2.3
345	2011-02-15 02:10:00	1158	2.2
7	1997-11-04 06:00:00	8100	2.1
318	2005-09-10 21:45:00	0808	2.1
420	2013-10-25 15:08:00	1882	2.1
361	2011-09-06 22:30:00	1283	2.1
274	2004-11-07 16:25:00	0696	2.0
98	2000-11-24 05:10:00	9236	2.0
125	2001-04-12 10:20:00	9415	2.0
285	2005-01-17 09:25:00	0720	2.0
102	2000-11-25 19:00:00	9236	1.9

Matching the Data:

Now let's at least find the flares that match up in both lists. As we can see below, all but 14 flares from the SWL data have immediate matches in the NASA flare data. The criteria we used to determine a 'match' is if at least two of the date, flare classification, and region are the same. The flares from the SWL data without an immediate NASA match are labeled as NaN for the time being.

In [195…

```
nasa_matches = []
taken_indeces = []

# Function to find a match in the NASA data
def find_nasa_match(swl_info, nasa_frame, match_parameter):
    swl_date = swl_info[0]
    swl_region = swl_info[1]
    swl_class = swl_info[2]

    # Iterate through NASA rows to find match
    for index, row in nasa_frame.iterrows():
        match_index = 0
        if row['start_datetime'].date() == swl_date:
            match_index += 1
        if row['flare_region'] == swl_region:
            match_index += 1

        try:
            if int(row['flare_classification']) == swl_class:
                match_index += 1
        except ValueError as e:
            match_index += 0

        # If suitable match found, return its index
        if match_index >= match_parameter and index not in taken_indeces:
            taken_indeces.append(index)
            return index

    # No match
    return np.nan


# Find the matching values
for index, row in swl_data.iterrows():
    swl_info = [row['start_datetime'].date(), row['region'], int(row['x_class
    match = find_nasa_match(swl_info, top_nasa_data, 2)

    nasa_matches.append(match)


matched_swl_data = updated_frame.copy()
matched_swl_data['nasa_index'] = nasa_matches

pd.set_option('display.max_rows', 50)
matched_swl_data
```

Out[195…

	rank	x_class	start_datetime	max_datetime	end_datetime	region	nasa_index
1	1	X28+	2003-11-04 19:29:00	2003-11-04 19:53:00	2003-11-04 20:06:00	0486	240.0
2	2	X20+	2001-04-02	2001-04-02	2001-04-02	9393	117.0

			21:32:00	21:51:00	22:03:00		
3	3	X17.2+	2003-10-28 09:51:00	2003-10-28 11:10:00	2003-10-28 11:24:00	0486	233.0
4	4	X17+	2005-09-07 17:17:00	2005-09-07 17:40:00	2005-09-07 18:03:00	0808	316.0
5	5	X14.4	2001-04-15 13:19:00	2001-04-15 13:50:00	2001-04-15 13:55:00	9415	126.0
6	6	X10	2003-10-29 20:37:00	2003-10-29 20:49:00	2003-10-29 21:01:00	0486	234.0
7	7	X9.4	1997-11-06 11:49:00	1997-11-06 11:55:00	1997-11-06 12:01:00	8100	8.0
8	8	X9.3	2017-09-06 11:53:00	2017-09-06 12:02:00	2017-09-06 12:10:00	2673	514.0
9	9	X9	2006-12-05 10:18:00	2006-12-05 10:35:00	2006-12-05 10:45:00	0930	328.0
10	10	X8.3	2003-11-02 17:03:00	2003-11-02 17:25:00	2003-11-02 17:39:00	0486	237.0
11	11	X8.2	2017-09-10 15:35:00	2017-09-10 16:06:00	2017-09-10 16:31:00	2673	515.0
12	12	X7.1	2005-01-20 06:36:00	2005-01-20 07:01:00	2005-01-20 07:26:00	0720	288.0
13	13	X6.9	2011-08-09 07:48:00	2011-08-09 08:05:00	2011-08-09 08:08:00	1263	359.0
14	14	X6.5	2006-12-06 18:29:00	2006-12-06 18:47:00	2006-12-06 19:00:00	0930	331.0
15	15	X6.2	2005-09-09 19:13:00	2005-09-09 20:04:00	2005-09-09 20:36:00	0808	317.0
16	16	X6.2	2001-12-13 14:20:00	2001-12-13 14:30:00	2001-12-13 14:35:00	9733	NaN
17	17	X5.7	2000-07-14 10:03:00	2000-07-14 10:24:00	2000-07-14 10:43:00	9077	82.0
18	18	X5.6	2001-04-06 19:10:00	2001-04-06 19:21:00	2001-04-06 19:31:00	9415	121.0
19	19	X5.4	2012-03-07 00:02:00	2012-03-07 00:24:00	2012-03-07 00:40:00	1429	375.0
20	20	X5.4	2005-09-08 20:52:00	2005-09-08 21:06:00	2005-09-08 21:17:00	0808	NaN
21	21	X5.4	2003-10-23 08:19:00	2003-10-23 08:35:00	2003-10-23 08:49:00	0486	NaN
22	22	X5.3	2001-08-25 16:23:00	2001-08-25 16:45:00	2001-08-25 17:04:00	9591	135.0

23	23	X4.9	2014-02-25 00:39:00	2014-02-25 00:49:00	2014-02-25 01:03:00	1990	443.0
24	24	X4.9	1998-08-18 22:10:00	1998-08-18 22:19:00	1998-08-18 22:28:00	8307	NaN
25	25	X4.8	2002-07-23 00:18:00	2002-07-23 00:35:00	2002-07-23 00:47:00	0039	193.0
26	26	X4	2000-11-26 16:34:00	2000-11-26 16:48:00	2000-11-26 16:56:00	9236	104.0
27	27	X3.9	2003-11-03 09:43:00	2003-11-03 09:55:00	2003-11-03 10:19:00	0488	239.0
28	28	X3.9	1998-08-19 21:35:00	1998-08-19 21:45:00	1998-08-19 21:50:00	8307	NaN
29	29	X3.8	2005-01-17 06:59:00	2005-01-17 09:52:00	2005-01-17 10:07:00	0720	286.0
30	30	X3.7	1998-11-22 06:30:00	1998-11-22 06:42:00	1998-11-22 06:49:00	8384	NaN
31	31	X3.6	2005-09-09 09:42:00	2005-09-09 09:59:00	2005-09-09 10:08:00	0808	NaN
32	32	X3.6	2004-07-16 13:49:00	2004-07-16 13:55:00	2004-07-16 14:01:00	0649	NaN
33	33	X3.6	2003-05-28 00:17:00	2003-05-28 00:27:00	2003-05-28 00:39:00	0365	222.0
34	34	X3.4	2006-12-13 02:14:00	2006-12-13 02:40:00	2006-12-13 02:57:00	0930	332.0
35	35	X3.4	2001-12-28 20:02:00	2001-12-28 20:45:00	2001-12-28 21:32:00	9767	160.0
36	36	X3.3	2013-11-05 22:07:00	2013-11-05 22:12:00	2013-11-05 22:15:00	1890	NaN
37	37	X3.3	2002-07-20 21:04:00	2002-07-20 21:30:00	2002-07-20 21:54:00	0039	192.0
38	38	X3.3	1998-11-28 04:54:00	1998-11-28 05:52:00	1998-11-28 06:13:00	8395	NaN
39	39	X3.2	2013-05-14 00:00:00	2013-05-14 01:11:00	2013-05-14 01:20:00	1748	404.0
40	40	X3.1	2014-10-24 21:07:00	2014-10-24 21:41:00	2014-10-24 22:13:00	2192	NaN
41	41	X3.1	2002-08-24 00:49:00	2002-08-24 01:12:00	2002-08-24 01:31:00	0069	201.0
42	42	X3	2002-07-15 19:59:00	2002-07-15 20:08:00	2002-07-15 20:14:00	0030	NaN

43	43	X2.8	2013-05-13 15:48:00	2013-05-13 16:05:00	2013-05-13 16:16:00	1748	403.0
44	44	X2.8	2001-12-11 07:58:00	2001-12-11 08:08:00	2001-12-11 08:14:00	9733	NaN
45	45	X2.8	1998-08-18 08:14:00	1998-08-18 08:24:00	1998-08-18 08:32:00	8307	NaN
46	46	X2.7	2015-05-05 22:05:00	2015-05-05 22:11:00	2015-05-05 22:15:00	2339	487.0
47	47	X2.7	2003-11-03 01:09:00	2003-11-03 01:30:00	2003-11-03 01:45:00	0488	238.0
48	48	X2.7	1998-05-06 07:58:00	1998-05-06 08:09:00	1998-05-06 08:20:00	8210	19.0
49	49	X2.6	2005-01-15 22:25:00	2005-01-15 23:02:00	2005-01-15 23:31:00	0720	284.0
50	50	X2.6	2001-09-24 09:32:00	2001-09-24 10:38:00	2001-09-24 11:09:00	9632	142.0

Searching the full NASA data:

Now, let's extract the missing rows from this dataset and attempt to find matches in the larger NASA set. However, since we know the region won't match up, first we can tighten our search to just the rows in the NASA set with flare classification unknown. We need to be less strict with our match conditions, since most flares on the NASA site with missing flare classifications also have missing flare regions, so our match function now will only require the dates matching up.

In [196…

```
nasa_missing_class = updated_nasa_frame[pd.isnull(updated_nasa_frame['flare_c
matched_swl_missing = matched_swl_data[pd.isnull(matched_swl_data['nasa_index

def update_nasa_index(swl_frame, nasa_frame, search_index):
    for index, row in swl_frame.iterrows():
        swl_info = [row['start_datetime'].date(), row['region'], int(remove_n
        match = find_nasa_match(swl_info, nasa_frame, search_index)

        if pd.isnull(swl_frame.at[index, 'nasa_index']):
            swl_frame.at[index, 'nasa_index'] = match

update_nasa_index(matched_swl_missing, nasa_missing_class, 1)
matched_swl_missing
```

Out[196…

	rank	x_class	start_datetime	max_datetime	end_datetime	region	nasa_index
16	16	X6.2	2001-12-13 14:20:00	2001-12-13 14:30:00	2001-12-13 14:35:00	9733	NaN
20	20	X5.4	2005-09-08 20:52:00	2005-09-08 21:06:00	2005-09-08 21:17:00	0808	NaN
21	21	X5.4	2003-10-23 08:19:00	2003-10-23 08:35:00	2003-10-23 08:49:00	0486	NaN
24	24	X4.9	1998-08-18 22:10:00	1998-08-18 22:19:00	1998-08-18 22:28:00	8307	NaN
28	28	X3.9	1998-08-19 21:35:00	1998-08-19 21:45:00	1998-08-19 21:50:00	8307	NaN
30	30	X3.7	1998-11-22 06:30:00	1998-11-22 06:42:00	1998-11-22 06:49:00	8384	NaN
31	31	X3.6	2005-09-09 09:42:00	2005-09-09 09:59:00	2005-09-09 10:08:00	0808	NaN
32	32	X3.6	2004-07-16 13:49:00	2004-07-16 13:55:00	2004-07-16 14:01:00	0649	NaN
36	36	X3.3	2013-11-05 22:07:00	2013-11-05 22:12:00	2013-11-05 22:15:00	1890	NaN
38	38	X3.3	1998-11-28 04:54:00	1998-11-28 05:52:00	1998-11-28 06:13:00	8395	NaN
40	40	X3.1	2014-10-24 21:07:00	2014-10-24 21:41:00	2014-10-24 22:13:00	2192	NaN
42	42	X3	2002-07-15 19:59:00	2002-07-15 20:08:00	2002-07-15 20:14:00	0030	NaN
44	44	X2.8	2001-12-11 07:58:00	2001-12-11 08:08:00	2001-12-11 08:14:00	9733	157.0
45	45	X2.8	1998-08-18 08:14:00	1998-08-18 08:24:00	1998-08-18 08:32:00	8307	NaN

Unfortunately, this only found one new match, so let's expand our search to the entire NASA dataset, in cases where the flare class wasn't missing data, but may just be different between the two sites.

In [197…

```
update_nasa_index(matched_swl_missing, updated_nasa_frame, 1)
matched_swl_missing
```

Out[197…

	rank	x_class	start_datetime	max_datetime	end_datetime	region	nasa_index
16	16	X6.2	2001-12-13 14:20:00	2001-12-13 14:30:00	2001-12-13 14:35:00	9733	NaN
20	20	X5.4	2005-09-08 20:52:00	2005-09-08 21:06:00	2005-09-08 21:17:00	0808	NaN
21	21	X5.4	2003-10-23 08:19:00	2003-10-23 08:35:00	2003-10-23 08:49:00	0486	NaN
24	24	X4.9	1998-08-18 22:10:00	1998-08-18 22:19:00	1998-08-18 22:28:00	8307	NaN
28	28	X3.9	1998-08-19 21:35:00	1998-08-19 21:45:00	1998-08-19 21:50:00	8307	NaN
30	30	X3.7	1998-11-22 06:30:00	1998-11-22 06:42:00	1998-11-22 06:49:00	8384	NaN
31	31	X3.6	2005-09-09 09:42:00	2005-09-09 09:59:00	2005-09-09 10:08:00	0808	NaN
32	32	X3.6	2004-07-16 13:49:00	2004-07-16 13:55:00	2004-07-16 14:01:00	0649	NaN
36	36	X3.3	2013-11-05 22:07:00	2013-11-05 22:12:00	2013-11-05 22:15:00	1890	NaN
38	38	X3.3	1998-11-28 04:54:00	1998-11-28 05:52:00	1998-11-28 06:13:00	8395	NaN
40	40	X3.1	2014-10-24 21:07:00	2014-10-24 21:41:00	2014-10-24 22:13:00	2192	NaN
42	42	X3	2002-07-15 19:59:00	2002-07-15 20:08:00	2002-07-15 20:14:00	0030	187.0
44	44	X2.8	2001-12-11 07:58:00	2001-12-11 08:08:00	2001-12-11 08:14:00	9733	157.0
45	45	X2.8	1998-08-18 08:14:00	1998-08-18 08:24:00	1998-08-18 08:32:00	8307	NaN

This helped us get a few more matches, but most are still left unfound. At this point, none of the dates, region, nor class are matching up, so we'll just have to find the closest match date-wise.

In [198…

```
def find_time_match(swl_row, nasa_frame):
    match = swl_row['nasa_index']

    # Iterate through NASA rows to find match
    for index, row in nasa_frame.iterrows():
        swl_date = swl_row['start_datetime']

        if index not in taken_indeces:
            if pd.isnull(match):
                taken_indeces.append(index)
                match = index
            else:
                time_difference = abs(row['start_datetime'] - swl_date)
                curr_time_difference = abs(nasa_frame.at[match, 'start_dateti
                if time_difference < curr_time_difference:
                    taken_indeces.append(index)
                    taken_indeces.remove(match)
                    match = index

    return match


for index, row in matched_swl_missing.iterrows():
    matched_swl_missing.at[index, 'nasa_index'] = find_time_match(row, update

matched_swl_missing
```

Out[198…

	rank	x_class	start_datetime	max_datetime	end_datetime	region	nasa_index
16	16	X6.2	2001-12-13 14:20:00	2001-12-13 14:30:00	2001-12-13 14:35:00	9733	158.0
20	20	X5.4	2005-09-08 20:52:00	2005-09-08 21:06:00	2005-09-08 21:17:00	0808	318.0
21	21	X5.4	2003-10-23 08:19:00	2003-10-23 08:35:00	2003-10-23 08:49:00	0486	230.0
24	24	X4.9	1998-08-18 22:10:00	1998-08-18 22:19:00	1998-08-18 22:28:00	8307	27.0
28	28	X3.9	1998-08-19 21:35:00	1998-08-19 21:45:00	1998-08-19 21:50:00	8307	26.0
30	30	X3.7	1998-11-22 06:30:00	1998-11-22 06:42:00	1998-11-22 06:49:00	8384	32.0
31	31	X3.6	2005-09-09 09:42:00	2005-09-09 09:59:00	2005-09-09 10:08:00	0808	319.0
32	32	X3.6	2004-07-16 13:49:00	2004-07-16 13:55:00	2004-07-16 14:01:00	0649	261.0
36	36	X3.3	2013-11-05 22:07:00	2013-11-05 22:12:00	2013-11-05 22:15:00	1890	427.0
38	38	X3.3	1998-11-28 04:54:00	1998-11-28 05:52:00	1998-11-28 06:13:00	8395	33.0
40	40	X3.1	2014-10-24 21:07:00	2014-10-24 21:41:00	2014-10-24 22:13:00	2192	474.0
42	42	X3	2002-07-15 19:59:00	2002-07-15 20:08:00	2002-07-15 20:14:00	0030	187.0
44	44	X2.8	2001-12-11 07:58:00	2001-12-11 08:08:00	2001-12-11 08:14:00	9733	157.0
45	45	X2.8	1998-08-18 08:14:00	1998-08-18 08:24:00	1998-08-18 08:32:00	8307	25.0

Completed SpaceWeatherLive Data

Now that we have found suitable matches for each of the flares, we can re-organize our data. Below is the now-filled in data from SpaceWeatherLive, with associated indeces for the matching NASA flares.

In [199…

```
for index, row in matched_swl_missing.iterrows():
    matched_swl_data.at[index, 'nasa_index'] = row['nasa_index']

matched_swl_data
```

Out[199…

	rank	x_class	start_datetime	max_datetime	end_datetime	region	nasa_index
1	1	X28+	2003-11-04 19:29:00	2003-11-04 19:53:00	2003-11-04 20:06:00	0486	240.0
2	2	X20+	2001-04-02 21:32:00	2001-04-02 21:51:00	2001-04-02 22:03:00	9393	117.0
3	3	X17.2+	2003-10-28 09:51:00	2003-10-28 11:10:00	2003-10-28 11:24:00	0486	233.0
4	4	X17+	2005-09-07 17:17:00	2005-09-07 17:40:00	2005-09-07 18:03:00	0808	316.0
5	5	X14.4	2001-04-15 13:19:00	2001-04-15 13:50:00	2001-04-15 13:55:00	9415	126.0
6	6	X10	2003-10-29 20:37:00	2003-10-29 20:49:00	2003-10-29 21:01:00	0486	234.0
7	7	X9.4	1997-11-06 11:49:00	1997-11-06 11:55:00	1997-11-06 12:01:00	8100	8.0
8	8	X9.3	2017-09-06 11:53:00	2017-09-06 12:02:00	2017-09-06 12:10:00	2673	514.0
9	9	X9	2006-12-05 10:18:00	2006-12-05 10:35:00	2006-12-05 10:45:00	0930	328.0
10	10	X8.3	2003-11-02 17:03:00	2003-11-02 17:25:00	2003-11-02 17:39:00	0486	237.0
11	11	X8.2	2017-09-10 15:35:00	2017-09-10 16:06:00	2017-09-10 16:31:00	2673	515.0
12	12	X7.1	2005-01-20 06:36:00	2005-01-20 07:01:00	2005-01-20 07:26:00	0720	288.0
13	13	X6.9	2011-08-09 07:48:00	2011-08-09 08:05:00	2011-08-09 08:08:00	1263	359.0
14	14	X6.5	2006-12-06 18:29:00	2006-12-06 18:47:00	2006-12-06 19:00:00	0930	331.0
15	15	X6.2	2005-09-09 19:13:00	2005-09-09 20:04:00	2005-09-09 20:36:00	0808	317.0
16	16	X6.2	2001-12-13 14:20:00	2001-12-13 14:30:00	2001-12-13 14:35:00	9733	158.0
17	17	X5.7	2000-07-14 10:03:00	2000-07-14 10:24:00	2000-07-14 10:43:00	9077	82.0
18	18	X5.6	2001-04-06 19:10:00	2001-04-06 19:21:00	2001-04-06 19:31:00	9415	121.0
19	19	X5.4	2012-03-07 00:02:00	2012-03-07 00:24:00	2012-03-07 00:40:00	1429	375.0
			2005-09-08	2005-09-08	2005-09-08		

20	20	X5.4	20:52:00	21:06:00	21:17:00	0808	318.0
21	21	X5.4	2003-10-23 08:19:00	2003-10-23 08:35:00	2003-10-23 08:49:00	0486	230.0
22	22	X5.3	2001-08-25 16:23:00	2001-08-25 16:45:00	2001-08-25 17:04:00	9591	135.0
23	23	X4.9	2014-02-25 00:39:00	2014-02-25 00:49:00	2014-02-25 01:03:00	1990	443.0
24	24	X4.9	1998-08-18 22:10:00	1998-08-18 22:19:00	1998-08-18 22:28:00	8307	27.0
25	25	X4.8	2002-07-23 00:18:00	2002-07-23 00:35:00	2002-07-23 00:47:00	0039	193.0
26	26	X4	2000-11-26 16:34:00	2000-11-26 16:48:00	2000-11-26 16:56:00	9236	104.0
27	27	X3.9	2003-11-03 09:43:00	2003-11-03 09:55:00	2003-11-03 10:19:00	0488	239.0
28	28	X3.9	1998-08-19 21:35:00	1998-08-19 21:45:00	1998-08-19 21:50:00	8307	26.0
29	29	X3.8	2005-01-17 06:59:00	2005-01-17 09:52:00	2005-01-17 10:07:00	0720	286.0
30	30	X3.7	1998-11-22 06:30:00	1998-11-22 06:42:00	1998-11-22 06:49:00	8384	32.0
31	31	X3.6	2005-09-09 09:42:00	2005-09-09 09:59:00	2005-09-09 10:08:00	0808	319.0
32	32	X3.6	2004-07-16 13:49:00	2004-07-16 13:55:00	2004-07-16 14:01:00	0649	261.0
33	33	X3.6	2003-05-28 00:17:00	2003-05-28 00:27:00	2003-05-28 00:39:00	0365	222.0
34	34	X3.4	2006-12-13 02:14:00	2006-12-13 02:40:00	2006-12-13 02:57:00	0930	332.0
35	35	X3.4	2001-12-28 20:02:00	2001-12-28 20:45:00	2001-12-28 21:32:00	9767	160.0
36	36	X3.3	2013-11-05 22:07:00	2013-11-05 22:12:00	2013-11-05 22:15:00	1890	427.0
37	37	X3.3	2002-07-20 21:04:00	2002-07-20 21:30:00	2002-07-20 21:54:00	0039	192.0
38	38	X3.3	1998-11-28 04:54:00	1998-11-28 05:52:00	1998-11-28 06:13:00	8395	33.0
39	39	X3.2	2013-05-14 00:00:00	2013-05-14 01:11:00	2013-05-14 01:20:00	1748	404.0
			2014-10-24	2014-10-24	2014-10-24		

40	40	X3.1	21:07:00	21:41:00	22:13:00	2192	474.0
41	41	X3.1	2002-08-24 00:49:00	2002-08-24 01:12:00	2002-08-24 01:31:00	0069	201.0
42	42	X3	2002-07-15 19:59:00	2002-07-15 20:08:00	2002-07-15 20:14:00	0030	187.0
43	43	X2.8	2013-05-13 15:48:00	2013-05-13 16:05:00	2013-05-13 16:16:00	1748	403.0
44	44	X2.8	2001-12-11 07:58:00	2001-12-11 08:08:00	2001-12-11 08:14:00	9733	157.0
45	45	X2.8	1998-08-18 08:14:00	1998-08-18 08:24:00	1998-08-18 08:32:00	8307	25.0
46	46	X2.7	2015-05-05 22:05:00	2015-05-05 22:11:00	2015-05-05 22:15:00	2339	487.0
47	47	X2.7	2003-11-03 01:09:00	2003-11-03 01:30:00	2003-11-03 01:45:00	0488	238.0
48	48	X2.7	1998-05-06 07:58:00	1998-05-06 08:09:00	1998-05-06 08:20:00	8210	19.0
49	49	X2.6	2005-01-15 22:25:00	2005-01-15 23:02:00	2005-01-15 23:31:00	0720	284.0
50	50	X2.6	2001-09-24 09:32:00	2001-09-24 10:38:00	2001-09-24 11:09:00	9632	142.0

Completed Top 50 Nasa Data

With these associated indeces, we can now rank our data scraped from the NASA site, making a list with the top 50 placed at the beginning

In [200…

```
swl_ranks = [np.nan] * 517
updated_nasa_frame['swl_rank'] = swl_ranks

for index, row in matched_swl_data.iterrows():
    nasa_index = row['nasa_index']
    updated_nasa_frame.at[nasa_index, 'swl_rank'] = index

updated_nasa_frame.sort_values(by='swl_rank')
```

Out[200…

	start_datetime	end_datetime	start_frequency	end_frequency	flare_location	flare_region
240	2003-11-04 20:00:00	2003-11-05 00:00:00	10000	200	S19W83	10486
117	2001-04-02 22:05:00	2001-04-03 02:30:00	14000	250	N19W72	9393
233	2003-10-28 11:10:00	2003-10-30 00:00:00	14000	40	S16E08	10486
316	2005-09-07 18:05:00	2005-09-08 00:00:00	12000	200	S11E77	10808
126	2001-04-15 14:05:00	2001-04-16 13:00:00	14000	40	S20W85	9415
...	...	...	...	...	...	...
511	2017-07-14 01:18:00	2017-07-14 21:30:00	14000	70	S06W29	12665
512	2017-07-23 05:27:00	2017-07-23 06:12:00	4400	900	BACK	NaN
513	2017-09-04 20:27:00	2017-09-05 04:54:00	14000	210	S10W12	12673
516	2017-09-12 07:38:00	2017-09-12 07:43:00	16000	13000	N08E48	12680
517	2017-09-17 11:45:00	2017-09-17 12:35:00	16000	900	S08E170	NaN

517 rows × 14 columns

Here's the data set limited to just the top 50 rows:

In [201…

```
pd.set_option('display.max_rows', 10)
updated_nasa_frame.sort_values(by='swl_rank')[:50]
```

Out[201…

	start_datetime	end_datetime	start_frequency	end_frequency	flare_location	flare_region
240	2003-11-04 20:00:00	2003-11-05 00:00:00	10000	200	S19W83	10486
117	2001-04-02 22:05:00	2001-04-03 02:30:00	14000	250	N19W72	9393
233	2003-10-28 11:10:00	2003-10-30 00:00:00	14000	40	S16E08	10486
316	2005-09-07 18:05:00	2005-09-08 00:00:00	12000	200	S11E77	10808
126	2001-04-15 14:05:00	2001-04-16 13:00:00	14000	40	S20W85	9415
…	…	…	…	…	…	…
487	2015-05-05 22:24:00	2015-05-05 23:14:00	14000	500	N15E79	12339
238	2003-11-03 01:15:00	2003-11-03 01:25:00	3000	1500	N10W83	10488
19	1998-05-06 08:25:00	1998-05-06 08:35:00	14000	5000	S11W65	8210
284	2005-01-15 23:00:00	2005-01-17 00:00:00	3000	40	N15W05	10720
142	2001-09-24 10:45:00	2001-09-25 20:00:00	7000	30	S16E23	9632

50 rows × 14 columns

Analysis: Are strong flares more likely in large clusters?

Now that we've collected our data, let's consider the following question: when a strong flare occurs (i.e., one in our top 50, for example), does it tend to occur alone, or in tandem with other flares? One way we might find out is to take a look at each time a strong flare occured, and see the number of flares that coincided with it. To consider this further, below is a bar graph showing each month a flare occured, and how many flares occured that month. The occurences of the top 50 flares are shown in red.

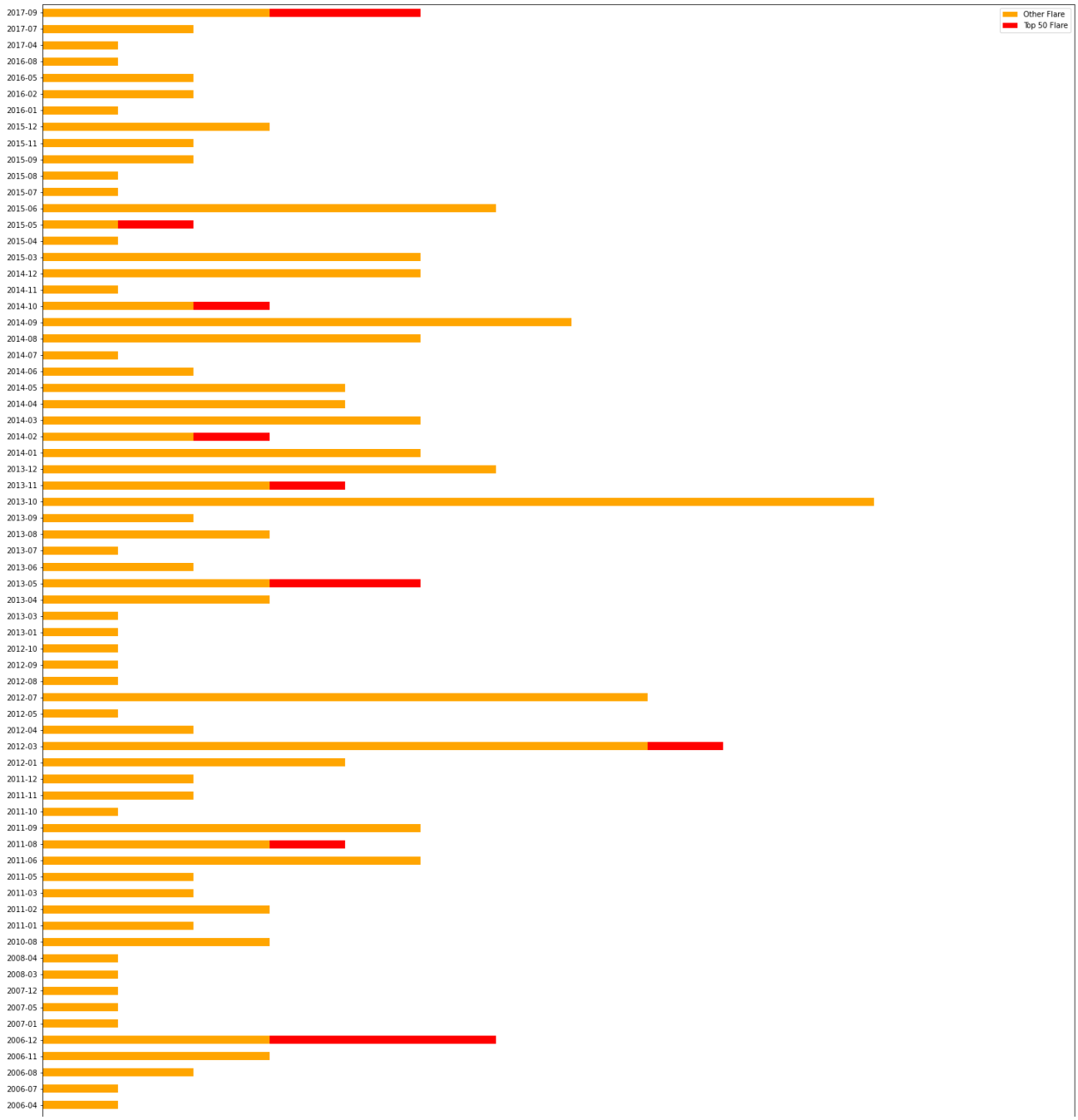
Note: months in which a flare did NOT occur are not shown

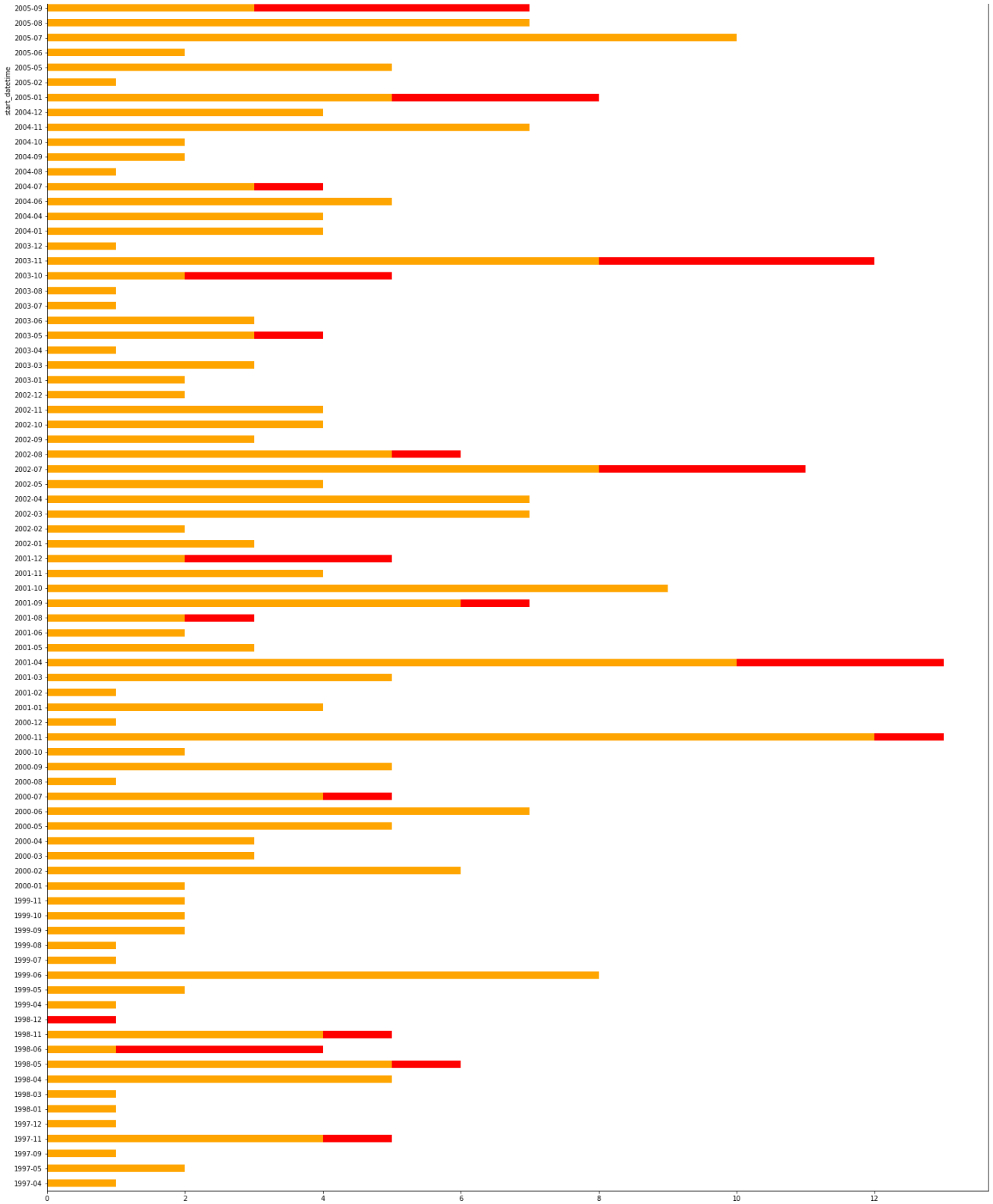
In [202…

```
import matplotlib.pyplot as plt

plot_nasa_frame = updated_nasa_frame.copy()

dates = plot_nasa_frame['start_datetime'].apply(lambda x: x.strftime('%Y-%m')
top_50 = plot_nasa_frame['swl_rank'].apply(lambda x: classify(x))
datesdf = dates.to_frame().join(top_50.to_frame())
bar = pd.crosstab(datesdf['start_datetime'], datesdf['swl_rank']).plot.barh(f
bar.legend(["Other Flare", "Top 50 Flare"])
bar
```

Out[202…

```
<AxesSubplot:ylabel='start_datetime'>
```




start_datetime
2005-09
2005-08
2005-07
2005-06
2005-05
2005-02
2005-01
2004-12
2004-11
2004-10
2004-09
2004-08
2004-07
2004-06
2004-04
2004-01
2003-12
2003-11
2003-10
2003-08
2003-07
2003-06
2003-05
2003-04
2003-03
2003-01
2002-12
2002-11
2002-10
2002-09
2002-08
2002-07
2002-05
2002-04
2002-03
2002-02
2002-01
2001-12
2001-11
2001-10
2001-09
2001-08
2001-06
2001-05
2001-04
2001-03
2001-02
2001-01
2000-12
2000-11
2000-10
2000-09
2000-08
2000-07
2000-06
2000-05
2000-04
2000-03
2000-02
2000-01
1999-11
1999-10
1999-09
1999-08
1999-07
1999-06
1999-05
1999-04
1998-12
1998-11
1998-06
1998-05
1998-04
1998-03
1998-01
1997-12
1997-11
1997-09
1997-05
1997-04
0
2
4
6
8
10
12

As we can see above, almost every top-50 flare occured in the same month as another flare, and only 5 top-50 flares occured in a month with fewer than 4 flares total. This would seem to suggest that strong flares do not occur alone. The single top-50 flare which did occur in a month all on its own, the one in December of 1998, is one of the flares that did not have a clear match in the SWL data, and is reported on the NASA site as only having been of class M, so it is a likely error in our dataset.

In [ ]: