CMSC320_P1

October 2, 2022

[1]:

```python
import requests, pandas, numpy
from bs4 import BeautifulSoup
from io import StringIO

### Part 1 Step 1

# Sent a get request and getting the HTML content from the website
# Parsed the HTML looking for the <table> tag
# Constructed a Pandas DataFrame using HTML with the data in the found table

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

bs = BeautifulSoup(req.content, "html.parser")
html_t = bs.find("table").prettify()

table = pandas.read_html(html_t)
df = table[0]
df.columns = ['rank', 'x_class', 'date', 'region', 'start_time', 'max_time',␣
 ↪'end_time', 'movie']
df
```

[1]:

```
    rank x_class        date  region start_time max_time end_time  \
0      1    X28+  2003/11/04     486      19:29    19:53    20:06
1      2    X20+  2001/04/02    9393      21:32    21:51    22:03
2      3  X17.2+  2003/10/28     486      09:51    11:10    11:24
3      4    X17+  2005/09/07     808      17:17    17:40    18:03
4      5   X14.4  2001/04/15    9415      13:19    13:50    13:55
5      6     X10  2003/10/29     486      20:37    20:49    21:01
6      7    X9.4  1997/11/06    8100      11:49    11:55    12:01
7      8    X9.3  2017/09/06    2673      11:53    12:02    12:10
8      9      X9  2006/12/05     930      10:18    10:35    10:45
9     10    X8.3  2003/11/02     486      17:03    17:25    17:39
10    11    X8.2  2017/09/10    2673      15:35    16:06    16:31
11    12    X7.1  2005/01/20     720      06:36    07:01    07:26
12    13    X6.9  2011/08/09    1263      07:48    08:05    08:08
13    14    X6.5  2006/12/06     930      18:29    18:47    19:00
14    15    X6.2  2005/09/09     808      19:13    20:04    20:36
```

```
15    16    X6.2  2001/12/13    9733      14:20    14:30    14:35
16    17    X5.7  2000/07/14    9077      10:03    10:24    10:43
17    18    X5.6  2001/04/06    9415      19:10    19:21    19:31
18    19    X5.4  2012/03/07    1429      00:02    00:24    00:40
19    20    X5.4  2005/09/08     808      20:52    21:06    21:17
20    21    X5.4  2003/10/23     486      08:19    08:35    08:49
21    22    X5.3  2001/08/25    9591      16:23    16:45    17:04
22    23    X4.9  2014/02/25    1990      00:39    00:49    01:03
23    24    X4.9  1998/08/18    8307      22:10    22:19    22:28
24    25    X4.8  2002/07/23      39      00:18    00:35    00:47
25    26      X4  2000/11/26    9236      16:34    16:48    16:56
26    27    X3.9  2003/11/03     488      09:43    09:55    10:19
27    28    X3.9  1998/08/19    8307      21:35    21:45    21:50
28    29    X3.8  2005/01/17     720      06:59    09:52    10:07
29    30    X3.7  1998/11/22    8384      06:30    06:42    06:49
30    31    X3.6  2005/09/09     808      09:42    09:59    10:08
31    32    X3.6  2004/07/16     649      13:49    13:55    14:01
32    33    X3.6  2003/05/28     365      00:17    00:27    00:39
33    34    X3.4  2006/12/13     930      02:14    02:40    02:57
34    35    X3.4  2001/12/28    9767      20:02    20:45    21:32
35    36    X3.3  2013/11/05    1890      22:07    22:12    22:15
36    37    X3.3  2002/07/20      39      21:04    21:30    21:54
37    38    X3.3  1998/11/28    8395      04:54    05:52    06:13
38    39    X3.2  2013/05/14    1748      00:00    01:11    01:20
39    40    X3.1  2014/10/24    2192      21:07    21:41    22:13
40    41    X3.1  2002/08/24      69      00:49    01:12    01:31
41    42      X3  2002/07/15      30      19:59    20:08    20:14
42    43    X2.8  2013/05/13    1748      15:48    16:05    16:16
43    44    X2.8  2001/12/11    9733      07:58    08:08    08:14
44    45    X2.8  1998/08/18    8307      08:14    08:24    08:32
45    46    X2.7  2015/05/05    2339      22:05    22:11    22:15
46    47    X2.7  2003/11/03     488      01:09    01:30    01:45
47    48    X2.7  1998/05/06    8210      07:58    08:09    08:20
48    49    X2.6  2005/01/15     720      22:25    23:02    23:31
49    50    X2.6  2001/09/24    9632      09:32    10:38    11:09

                  movie
0   Movie  View archive
1   Movie  View archive
2   Movie  View archive
3   Movie  View archive
4   Movie  View archive
5   Movie  View archive
6   Movie  View archive
7   Movie  View archive
8   Movie  View archive
9   Movie  View archive
```

```
10  Movie  View archive
11  Movie  View archive
12  Movie  View archive
13  Movie  View archive
14  Movie  View archive
15  Movie  View archive
16  Movie  View archive
17  Movie  View archive
18  Movie  View archive
19  Movie  View archive
20  Movie  View archive
21  Movie  View archive
22  Movie  View archive
23         View archive
24  Movie  View archive
25  Movie  View archive
26  Movie  View archive
27         View archive
28  Movie  View archive
29  Movie  View archive
30  Movie  View archive
31  Movie  View archive
32  Movie  View archive
33  Movie  View archive
34  Movie  View archive
35  Movie  View archive
36  Movie  View archive
37  Movie  View archive
38  Movie  View archive
39  Movie  View archive
40  Movie  View archive
41  Movie  View archive
42  Movie  View archive
43  Movie  View archive
44         View archive
45  Movie  View archive
46  Movie  View archive
47  Movie  View archive
48  Movie  View archive
49  Movie  View archive
```

[2]:

```
### Part 1 Step 2

# Iterate through the DataFrame combining the date with the time to form timestamps
# Set column types to datetime64
# Replaced missing data with 'null' placeholder
# Rearranged DataFrame columns

data = df.drop('movie', axis=1).copy()

for index, row in data.iterrows():
  data.iat[index, 4] = pandas.to_datetime(row['date'] + ' ' + row['start_time'])
  data.iat[index, 5] = pandas.to_datetime(row['date'] + ' ' + row['max_time'])
  data.iat[index, 6] = pandas.to_datetime(row['date'] + ' ' + row['end_time'])

data = data.drop('date', axis=1)

data['start_time'] = data['start_time'].astype('datetime64')
data['max_time'] = data['max_time'].astype('datetime64')
data['end_time'] = data['end_time'].astype('datetime64')

data.replace('-', 'null')

data = data[['rank', 'x_class', 'start_time', 'max_time', 'end_time', 'region']]
data
```

[2]:

```
    rank x_class          start_time            max_time            end_time  \
0      1    X28+ 2003-11-04 19:29:00 2003-11-04 19:53:00 2003-11-04 20:06:00
1      2    X20+ 2001-04-02 21:32:00 2001-04-02 21:51:00 2001-04-02 22:03:00
2      3  X17.2+ 2003-10-28 09:51:00 2003-10-28 11:10:00 2003-10-28 11:24:00
3      4    X17+ 2005-09-07 17:17:00 2005-09-07 17:40:00 2005-09-07 18:03:00
4      5   X14.4 2001-04-15 13:19:00 2001-04-15 13:50:00 2001-04-15 13:55:00
5      6     X10 2003-10-29 20:37:00 2003-10-29 20:49:00 2003-10-29 21:01:00
6      7    X9.4 1997-11-06 11:49:00 1997-11-06 11:55:00 1997-11-06 12:01:00
7      8    X9.3 2017-09-06 11:53:00 2017-09-06 12:02:00 2017-09-06 12:10:00
8      9      X9 2006-12-05 10:18:00 2006-12-05 10:35:00 2006-12-05 10:45:00
9     10    X8.3 2003-11-02 17:03:00 2003-11-02 17:25:00 2003-11-02 17:39:00
10    11    X8.2 2017-09-10 15:35:00 2017-09-10 16:06:00 2017-09-10 16:31:00
11    12    X7.1 2005-01-20 06:36:00 2005-01-20 07:01:00 2005-01-20 07:26:00
12    13    X6.9 2011-08-09 07:48:00 2011-08-09 08:05:00 2011-08-09 08:08:00
13    14    X6.5 2006-12-06 18:29:00 2006-12-06 18:47:00 2006-12-06 19:00:00
14    15    X6.2 2005-09-09 19:13:00 2005-09-09 20:04:00 2005-09-09 20:36:00
15    16    X6.2 2001-12-13 14:20:00 2001-12-13 14:30:00 2001-12-13 14:35:00
16    17    X5.7 2000-07-14 10:03:00 2000-07-14 10:24:00 2000-07-14 10:43:00
17    18    X5.6 2001-04-06 19:10:00 2001-04-06 19:21:00 2001-04-06 19:31:00
18    19    X5.4 2012-03-07 00:02:00 2012-03-07 00:24:00 2012-03-07 00:40:00
19    20    X5.4 2005-09-08 20:52:00 2005-09-08 21:06:00 2005-09-08 21:17:00
20    21    X5.4 2003-10-23 08:19:00 2003-10-23 08:35:00 2003-10-23 08:49:00
21    22    X5.3 2001-08-25 16:23:00 2001-08-25 16:45:00 2001-08-25 17:04:00
22    23    X4.9 2014-02-25 00:39:00 2014-02-25 00:49:00 2014-02-25 01:03:00
23    24    X4.9 1998-08-18 22:10:00 1998-08-18 22:19:00 1998-08-18 22:28:00
24    25    X4.8 2002-07-23 00:18:00 2002-07-23 00:35:00 2002-07-23 00:47:00
25    26      X4 2000-11-26 16:34:00 2000-11-26 16:48:00 2000-11-26 16:56:00
```

```
26    27    X3.9 2003-11-03 09:43:00 2003-11-03 09:55:00 2003-11-03 10:19:00
27    28    X3.9 1998-08-19 21:35:00 1998-08-19 21:45:00 1998-08-19 21:50:00
28    29    X3.8 2005-01-17 06:59:00 2005-01-17 09:52:00 2005-01-17 10:07:00
29    30    X3.7 1998-11-22 06:30:00 1998-11-22 06:42:00 1998-11-22 06:49:00
30    31    X3.6 2005-09-09 09:42:00 2005-09-09 09:59:00 2005-09-09 10:08:00
31    32    X3.6 2004-07-16 13:49:00 2004-07-16 13:55:00 2004-07-16 14:01:00
32    33    X3.6 2003-05-28 00:17:00 2003-05-28 00:27:00 2003-05-28 00:39:00
33    34    X3.4 2006-12-13 02:14:00 2006-12-13 02:40:00 2006-12-13 02:57:00
34    35    X3.4 2001-12-28 20:02:00 2001-12-28 20:45:00 2001-12-28 21:32:00
35    36    X3.3 2013-11-05 22:07:00 2013-11-05 22:12:00 2013-11-05 22:15:00
36    37    X3.3 2002-07-20 21:04:00 2002-07-20 21:30:00 2002-07-20 21:54:00
37    38    X3.3 1998-11-28 04:54:00 1998-11-28 05:52:00 1998-11-28 06:13:00
38    39    X3.2 2013-05-14 00:00:00 2013-05-14 01:11:00 2013-05-14 01:20:00
39    40    X3.1 2014-10-24 21:07:00 2014-10-24 21:41:00 2014-10-24 22:13:00
40    41    X3.1 2002-08-24 00:49:00 2002-08-24 01:12:00 2002-08-24 01:31:00
41    42      X3 2002-07-15 19:59:00 2002-07-15 20:08:00 2002-07-15 20:14:00
42    43    X2.8 2013-05-13 15:48:00 2013-05-13 16:05:00 2013-05-13 16:16:00
43    44    X2.8 2001-12-11 07:58:00 2001-12-11 08:08:00 2001-12-11 08:14:00
44    45    X2.8 1998-08-18 08:14:00 1998-08-18 08:24:00 1998-08-18 08:32:00
45    46    X2.7 2015-05-05 22:05:00 2015-05-05 22:11:00 2015-05-05 22:15:00
46    47    X2.7 2003-11-03 01:09:00 2003-11-03 01:30:00 2003-11-03 01:45:00
47    48    X2.7 1998-05-06 07:58:00 1998-05-06 08:09:00 1998-05-06 08:20:00
48    49    X2.6 2005-01-15 22:25:00 2005-01-15 23:02:00 2005-01-15 23:31:00
49    50    X2.6 2001-09-24 09:32:00 2001-09-24 10:38:00 2001-09-24 11:09:00

    region
0      486
1     9393
2      486
3      808
4     9415
5      486
6     8100
7     2673
8      930
9      486
10    2673
11     720
12    1263
13     930
14     808
15    9733
16    9077
17    9415
18    1429
19     808
20     486
```

```
21    9591
22    1990
23    8307
24      39
25    9236
26     488
27    8307
28     720
29    8384
30     808
31     649
32     365
33     930
34    9767
35    1890
36      39
37    8395
38    1748
39    2192
40      69
41      30
42    1748
43    9733
44    8307
45    2339
46     488
47    8210
48     720
49    9632
```

[5]:

```python
### Part 1 Step 3

# Sent a get request and getting the HTML content from the website
# Parsed the HTML looking for the <pre> tag where the data is stored
# Split content by rows into array and trim the array to remove unnecessary␣
 ↪lines leaving only the raw data
# Removed the unncessary extra text after the PHTX column
# Converted the array into a CSV format using whitespace as the delimiter so␣
 ↪that it can be read by pandas using .read_csv() to for the DataFrame

nasa_req = requests.get('https://cmsc320.github.io/files/waves_type2.html')

nasa_bs = BeautifulSoup(nasa_req.content, 'lxml')
nasa_html = nasa_bs.find('pre').contents[0]

nasa_rows = nasa_bs.get_text().split('\n')
nasa_rows = nasa_rows[15: len(nasa_rows) - 3]
```

```
for i in range(len(nasa_rows)):
  nasa_rows[i] = nasa_rows[i].split('PHTX', 1)[0]

csv = 'start_date start_time end_date end_time start_frequency end_frequency␣
 ↪flare_location flare_region flare_classification cme_date cme_time cme_angle␣
 ↪cme_width cme_speed'
for i in range(len(nasa_rows)):
  csv = csv + '\n' + nasa_rows[i]

nasa_df = pandas.read_csv(StringIO(csv), delim_whitespace=True)

nasa_df
```

[5]:

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

    flare_location flare_region flare_classification cme_date cme_time  \
0           S25E16         8026                 M1.3    04/01    15:18
1           S28E19         8027                 C6.8    04/07    14:27
2           N21W08         8038                 C1.3    05/12    05:30
3           N05W12         8040                 M1.3    05/21    21:00
4           S29E25         8088                 C1.4    09/23    22:02
..             …          …                  …      …      …
513         S10W12        12673                 M5.5    09/04    20:12
514         S08W33        12673                 X9.3    09/06    12:24
515         S09W92        -----                 X8.3    09/10    16:00
516         N08E48        12680                 C3.0    09/12    08:03
517        S08E170        -----                 ----    09/17    12:00

    cme_angle cme_width cme_speed
0          74        79       312
1        Halo       360       878
2        Halo       360       464
3         263       165       296
4         133       155       712
..        …       …       …
```

```
513      Halo      360      1418
514      Halo      360      1571
515      Halo      360      3163
516       124       96       252
517      Halo      360      1385

[518 rows x 14 columns]
```

[4]:

```
### Part 1 Step 4

# Replaced the missing data placeholders with 'NaN'
# Go through each row of the dataset to check whether or no there is a Halo or␣
 ↪lower bound storing the result in two lists.
# Adds the resulting lists as new columns.

# Combined the date and time for 'start_datetime', 'end_datetime',␣
 ↪'cme_datetime'
# In the case of 'end_datetime', some values in 'end_time' are '24:00" which is␣
 ↪not an acceptable format for Pandas' datetime datattype.
# As such the '24:00' is converted to '00:00' and the date is incremented by a␣
 ↪day.

# Dropped the unnecessary columns and renames the remaining columns.
# Set type of date column, did not cast 'cme_datetime' to datetime type as some␣
 ↪rows have 'NaN'.

nasa_data = nasa_df.copy()

nasa_data = nasa_data.replace('-----', 'NaN')
nasa_data = nasa_data.replace('----', 'NaN')
nasa_data = nasa_data.replace('--/--', 'NaN')
nasa_data = nasa_data.replace('--:--', 'NaN')
nasa_data = nasa_data.replace('????', 'NaN')
nasa_data = nasa_data.replace('------', 'NaN')
nasa_data = nasa_data.replace('BACK', 'NaN')
nasa_data = nasa_data.replace('Back', 'NaN')

halo = []
for i, row in nasa_data.iterrows():
  if row['cme_angle'] == 'Halo':
    halo.append(True)
    nasa_data.at[i, 'cme_angle'] = 'NaN'
  else:
    halo.append(False)

nasa_data.insert(14, 'is_halo', halo, True)
```

```
lower_bound = []
for i, row in nasa_data.iterrows():
  if '>' in row['cme_width']:
    lower_bound.append(True)
    nasa_data.at[i, 'cme_width'] = row['cme_width'][1:len(row['cme_width'])]
  else:
    lower_bound.append(False)

nasa_data.insert(15, 'is_lower_bound', lower_bound, True)

for i, row in nasa_data.iterrows():
  nasa_data.at[i, 'start_time'] = pandas.to_datetime(row['start_date'] + ' ' +␣
 ↪row['start_time'])

  if row['end_time'] == '24:00':
    nasa_data.at[i, 'end_time'] = pandas.to_datetime(row['start_date'][0:4] + '/
 ↪' + row['end_date'] + ' ' + '00:00') + pandas.DateOffset(1)
  else:
    nasa_data.at[i, 'end_time'] = pandas.to_datetime(row['start_date'][0:4] + '/
 ↪' + row['end_date'] + ' ' + row['end_time'])

  if row['cme_date'] != 'NaN':
    nasa_data.at[i, 'cme_time'] = pandas.to_datetime(row['start_date'][0:4] + '/
 ↪' + row['cme_date'] + ' ' + row['cme_time'])

nasa_data = nasa_data.drop(columns=['start_date', 'end_date', 'cme_date'])
nasa_data = nasa_data.rename(columns={'start_time': 'start_datetime',␣
 ↪'end_time': 'end_datetime', 'cme_time': 'cme_datetime'})

nasa_data['start_datetime'] = nasa_data['start_datetime'].astype('datetime64')
nasa_data['end_datetime'] = nasa_data['end_datetime'].astype('datetime64')

nasa_data
```

[4]:

```
         start_datetime        end_datetime start_frequency end_frequency  \
0   1997-04-01 14:00:00 1997-04-01 14:15:00            8000          4000
1   1997-04-07 14:30:00 1997-04-07 17:30:00           11000          1000
2   1997-05-12 05:15:00 1997-05-14 16:00:00           12000            80
3   1997-05-21 20:20:00 1997-05-21 22:00:00            5000           500
4   1997-09-23 21:53:00 1997-09-23 22:16:00            6000          2000
..                  …                   …               …             …
513 2017-09-04 20:27:00 2017-09-05 04:54:00           14000           210
514 2017-09-06 12:05:00 2017-09-07 08:00:00           16000            70
515 2017-09-10 16:02:00 2017-09-11 06:50:00           16000           150
516 2017-09-12 07:38:00 2017-09-12 07:43:00           16000         13000
517 2017-09-17 11:45:00 2017-09-17 12:35:00           16000           900
```

```
    flare_location flare_region flare_classification         cme_datetime  \
0           S25E16         8026                 M1.3  1997-04-01 15:18:00
1           S28E19         8027                 C6.8  1997-04-07 14:27:00
2           N21W08         8038                 C1.3  1997-05-12 05:30:00
3           N05W12         8040                 M1.3  1997-05-21 21:00:00
4           S29E25         8088                 C1.4  1997-09-23 22:02:00
..             …          …                  …                   …
513         S10W12        12673                 M5.5  2017-09-04 20:12:00
514         S08W33        12673                 X9.3  2017-09-06 12:24:00
515         S09W92          NaN                 X8.3  2017-09-10 16:00:00
516         N08E48        12680                 C3.0  2017-09-12 08:03:00
517        S08E170          NaN                  NaN  2017-09-17 12:00:00

    cme_angle cme_width cme_speed  is_halo  is_lower_bound
0          74        79       312    False           False
1         NaN       360       878     True           False
2         NaN       360       464     True           False
3         263       165       296    False           False
4         133       155       712    False           False
..        …       …       …      …             …
513       NaN       360      1418     True           False
514       NaN       360      1571     True           False
515       NaN       360      3163     True           False
516       124        96       252    False           False
517       NaN       360      1385     True           False

[518 rows x 13 columns]
```

[10]:

```
### Part 2 Question 1

# Filters the DataFrame for all flares with 'X' in its classification.
# Creates a temporary table extracting the numerical value after the 'X' in 'flare_classification' and sorts the temporary column by descending order.
# Drop the temporary columns and take the top 50.

sf50 = nasa_data[nasa_data['flare_classification'].str.contains('X')].copy()
sf50['temp'] = sf50['flare_classification']
sf50['temp'] = sf50['temp'].apply(lambda x: float(x[1:]))
sf50.sort_values(['temp'], ascending=False, inplace=True)
sf50 = sf50.drop('temp', axis=1).head(50)

sf50
```

[10]:

```
         start_datetime        end_datetime  start_frequency end_frequency  \
240 2003-11-04 20:00:00 2003-11-05 00:00:00            10000           200
117 2001-04-02 22:05:00 2001-04-03 02:30:00            14000           250
233 2003-10-28 11:10:00 2003-10-30 00:00:00            14000            40
```

126	2001-04-15 14:05:00	2001-04-16 13:00:00	14000	40
234	2003-10-29 20:55:00	2003-10-30 00:00:00	11000	500
8	1997-11-06 12:20:00	1997-11-07 08:30:00	14000	100
514	2017-09-06 12:05:00	2017-09-07 08:00:00	16000	70
328	2006-12-05 10:50:00	2006-12-05 20:00:00	14000	250
237	2003-11-02 17:30:00	2003-11-03 01:00:00	12000	250
515	2017-09-10 16:02:00	2017-09-11 06:50:00	16000	150
288	2005-01-20 07:15:00	2005-01-20 16:30:00	14000	25
359	2011-08-09 08:20:00	2011-08-09 08:35:00	16000	4000
331	2006-12-06 19:00:00	2006-12-09 00:00:00	16000	30
317	2005-09-09 19:45:00	2005-09-09 22:00:00	10000	50
82	2000-07-14 10:30:00	2000-07-15 14:30:00	14000	80
121	2001-04-06 19:35:00	2001-04-07 01:50:00	14000	230
375	2012-03-07 01:00:00	2012-03-08 19:00:00	16000	30
135	2001-08-25 16:50:00	2001-08-25 23:00:00	8000	170
443	2014-02-25 00:56:00	2014-02-25 11:28:00	14000	100
193	2002-07-23 00:50:00	2002-07-23 04:00:00	11000	400
104	2000-11-26 17:00:00	2000-11-26 17:15:00	14000	7000
239	2003-11-03 10:00:00	2003-11-03 12:30:00	6000	400
286	2005-01-17 10:00:00	2005-01-17 10:35:00	6100	1500
222	2003-05-28 01:00:00	2003-05-29 00:30:00	1000	200
332	2006-12-13 02:45:00	2006-12-13 10:40:00	12000	150
160	2001-12-28 20:35:00	2001-12-29 03:00:00	14000	350
192	2002-07-20 21:30:00	2002-07-20 22:20:00	10000	2000
404	2013-05-14 01:16:00	2013-05-14 08:20:00	16000	240
201	2002-08-24 01:45:00	2002-08-24 03:25:00	5000	400
403	2013-05-13 16:15:00	2013-05-13 19:10:00	16000	300
487	2015-05-05 22:24:00	2015-05-05 23:14:00	14000	500
19	1998-05-06 08:25:00	1998-05-06 08:35:00	14000	5000
238	2003-11-03 01:15:00	2003-11-03 01:25:00	3000	1500
284	2005-01-15 23:00:00	2005-01-17 00:00:00	3000	40
142	2001-09-24 10:45:00	2001-09-25 20:00:00	7000	30
9	1997-11-27 13:30:00	1997-11-27 14:00:00	14000	7000
276	2004-11-10 02:25:00	2004-11-10 03:40:00	14000	1000
123	2001-04-10 05:24:00	2001-04-11 00:00:00	14000	100
99	2000-11-24 15:25:00	2000-11-24 22:00:00	14000	200
73	2000-06-06 15:20:00	2000-06-08 09:00:00	14000	40
345	2011-02-15 02:10:00	2011-02-15 07:00:00	16000	400
318	2005-09-10 21:45:00	2005-09-11 01:00:00	14000	200
361	2011-09-06 22:30:00	2011-09-07 15:40:00	16000	150
420	2013-10-25 15:08:00	2013-10-25 22:32:00	16000	200
7	1997-11-04 06:00:00	1997-11-05 04:30:00	14000	100
98	2000-11-24 05:10:00	2000-11-24 15:00:00	14000	100
125	2001-04-12 10:20:00	2001-04-12 10:40:00	14000	7000
274	2004-11-07 16:25:00	2004-11-08 20:00:00	14000	60
285	2005-01-17 09:25:00	2005-01-17 16:00:00	14000	30
102	2000-11-25 19:00:00	2000-11-25 19:35:00	6000	2000

```
    flare_location flare_region flare_classification         cme_datetime  \
240         S19W83        10486                 X28.  2003-11-04 19:54:00
117         N19W72         9393                 X20.  2001-04-02 22:06:00
233         S16E08        10486                 X17.  2003-10-28 11:30:00
126         S20W85         9415                 X14.  2001-04-15 14:06:00
234         S15W02        10486                 X10.  2003-10-29 20:54:00
8           S18W63         8100                 X9.4  1997-11-06 12:10:00
514         S08W33        12673                 X9.3  2017-09-06 12:24:00
328         S07E68        10930                 X9.0                  NaN
237         S14W56        10486                 X8.3  2003-11-02 17:30:00
515         S09W92          NaN                 X8.3  2017-09-10 16:00:00
288         N14W61        10720                 X7.1  2005-01-20 06:54:00
359         N17W69        11263                 X6.9  2011-08-09 08:12:00
331         S05E64        10930                 X6.5                  NaN
317         S12E67        10808                 X6.2  2005-09-09 19:48:00
82          N22W07         9077                 X5.7  2000-07-14 10:54:00
121         S21E31         9415                 X5.6  2001-04-06 19:30:00
375         N17E27        11429                 X5.4  2012-03-07 00:24:00
135         S17E34         9591                 X5.3  2001-08-25 16:50:00
443         S12E82        11990                 X4.9  2014-02-25 01:25:00
193         S13E72        10039                 X4.8  2002-07-23 00:42:00
104         N18W38         9236                 X4.0  2000-11-26 17:06:00
239         N08W77        10488                 X3.9  2003-11-03 10:06:00
286         N15W25        10720                 X3.8  2005-01-17 09:54:00
222         S07W20        10365                 X3.6  2003-05-28 00:50:00
332         S06W23        10930                 X3.4  2006-12-13 02:54:00
160         S26E90         9756                 X3.4  2001-12-28 20:30:00
192         S13E90        10039                 X3.3  2002-07-20 22:06:00
404         N08E77        11748                 X3.2  2013-05-14 01:25:00
201         S02W81        10069                 X3.1  2002-08-24 01:27:00
403         N11E85        11748                 X2.8  2013-05-13 16:07:00
487         N15E79        12339                 X2.7  2015-05-05 22:24:00
19          S11W65         8210                 X2.7  1998-05-06 08:29:00
238         N10W83        10488                 X2.7  2003-11-03 01:59:00
284         N15W05        10720                 X2.6  2005-01-15 23:06:00
142         S16E23         9632                 X2.6  2001-09-24 10:30:00
9           N17E63         8113                 X2.6  1997-11-27 13:56:00
276         N09W49        10696                 X2.5  2004-11-10 02:26:00
123         S23W09         9415                 X2.3  2001-04-10 05:30:00
99          N22W07         9236                 X2.3  2000-11-24 15:30:00
73          N20E18         9026                 X2.3  2000-06-06 15:54:00
345         S20W12        11158                 X2.2  2011-02-15 02:24:00
318         S13E47        10808                 X2.1  2005-09-10 21:52:00
361         N14W18        11283                 X2.1  2011-09-06 23:05:00
420         S06E69        11882                 X2.1  2013-10-25 15:12:00
7           S14W33         8100                 X2.1  1997-11-04 06:10:00
```

```
98          N20W05         9236                 X2.0  2000-11-24 05:30:00
125         S19W43         9415                 X2.0  2001-04-12 10:31:00
274         N09W17        10696                 X2.0  2004-11-07 16:54:00
285         N15W25        10720                 X2.0  2005-01-17 09:30:00
102         N20W23         9236                 X1.9  2000-11-25 19:31:00

    cme_angle cme_width cme_speed  is_halo  is_lower_bound
240       NaN       360      2657     True           False
117       261       244      2505    False           False
233       NaN       360      2459     True           False
126       245       167      1199    False           False
234       NaN       360      2029     True           False
8         NaN       360      1556     True           False
514       NaN       360      1571     True           False
328       NaN       NaN       NaN    False           False
237       NaN       360      2598     True           False
515       NaN       360      3163     True           False
288       NaN       360       882     True           False
359       NaN       360      1610     True           False
331       NaN       NaN       NaN    False           False
317       NaN       360      2257     True           False
82        NaN       360      1674     True           False
121       NaN       360      1270     True           False
375       NaN       360      2684     True           False
135       NaN       360      1433     True           False
443       NaN       360      2147     True           False
193       NaN       360      2285     True           False
104       NaN       360       980     True           False
239       293       103      1420    False           False
286       NaN       360      2547     True           False
222       NaN       360      1366     True           False
332       NaN       360      1774     True           False
160       NaN       360      2216     True           False
192       NaN       360      1941     True           False
404       NaN       360      2625     True           False
201       NaN       360      1913     True           False
403       NaN       360      1850     True           False
487       NaN       360       715     True           False
19        309       190      1099    False           False
238       304        65       827    False           False
284       NaN       360      2861     True           False
142       NaN       360      2402     True           False
9          98        91       441    False           False
276       NaN       360      3387     True           False
123       NaN       360      2411     True           False
99        NaN       360      1245     True           False
73        NaN       360      1119     True           False
```

```
345       NaN       360        669     True           False
318       NaN       360       1893     True           False
361       NaN       360        575     True           False
420       NaN       360       1081     True           False
7         NaN       360        785     True           False
98        NaN       360       1289     True           False
125       NaN       360       1184     True           False
274       NaN       360       1759     True           False
285       NaN       360       2094     True           False
102       NaN       360        671     True           False
```

I was not able to replicate SpaceWeatherLive's data exactly row for row. The first few rows of my output were similar to SpaceWeatherLive's data, but as I went down the list I noticed my data produced a minumum classification of X1.9 whereas their data stopped at X2.6.

[8]:

```
### Part 2 Question 2

# Created a temporary column 'x_class' for the top 50 data that was obtained as␣
 ↪a result from Part 2 Question 1.
# Removed all occurences of 'X' and '+' leaving only the numerical value in the␣
 ↪'x_class' column for both SpaceWeatherLive's and NASA's Top 50's data.
# Converted SpaceWeatherLive's region code following the pattern in the text␣
 ↪explanation below.
# Created a new 'rank' column, matched the two datasets, and set value inside␣
 ↪'rank' column according matched rows.

match_data = sf50.copy()
match_data['x_class'] = match_data['flare_classification'].apply(lambda x:␣
 ↪float(x[1:]))
sf_temp = data.copy()
sf_temp['x_class'] = sf_temp['x_class'].apply(lambda x: x[:-1] if ('+' in x)␣
 ↪else x)
sf_temp['x_class'] = sf_temp['x_class'].apply(lambda x: float(x[1:]))
sf_temp['region'] = sf_temp['region'].apply(lambda x: '10' + str(x) if␣
 ↪len(str(x)) == 3 else ('100' + str(x) if len(str(x)) == 2 else x))
match_data['rank'] = ''

for i, nasa_row in match_data.iterrows():
  for j, sf_row in sf_temp.iterrows():
    if str(match_data.at[i, 'x_class']) == str(sf_temp.at[j, 'x_class']) and \
    str(match_data.at[i, 'flare_region']) == str(sf_temp.at[j, 'region']) and \
    pandas.to_datetime(match_data.at[i, 'start_datetime']).normalize() ==␣
 ↪pandas.to_datetime(sf_temp.at[j, 'start_time']).normalize():
      match_data.at[i, 'rank'] = sf_temp.at[j, 'rank']

match_data = match_data.drop('x_class', axis=1)
match_data[match_data['rank'] != ''].sort_values('rank')
```

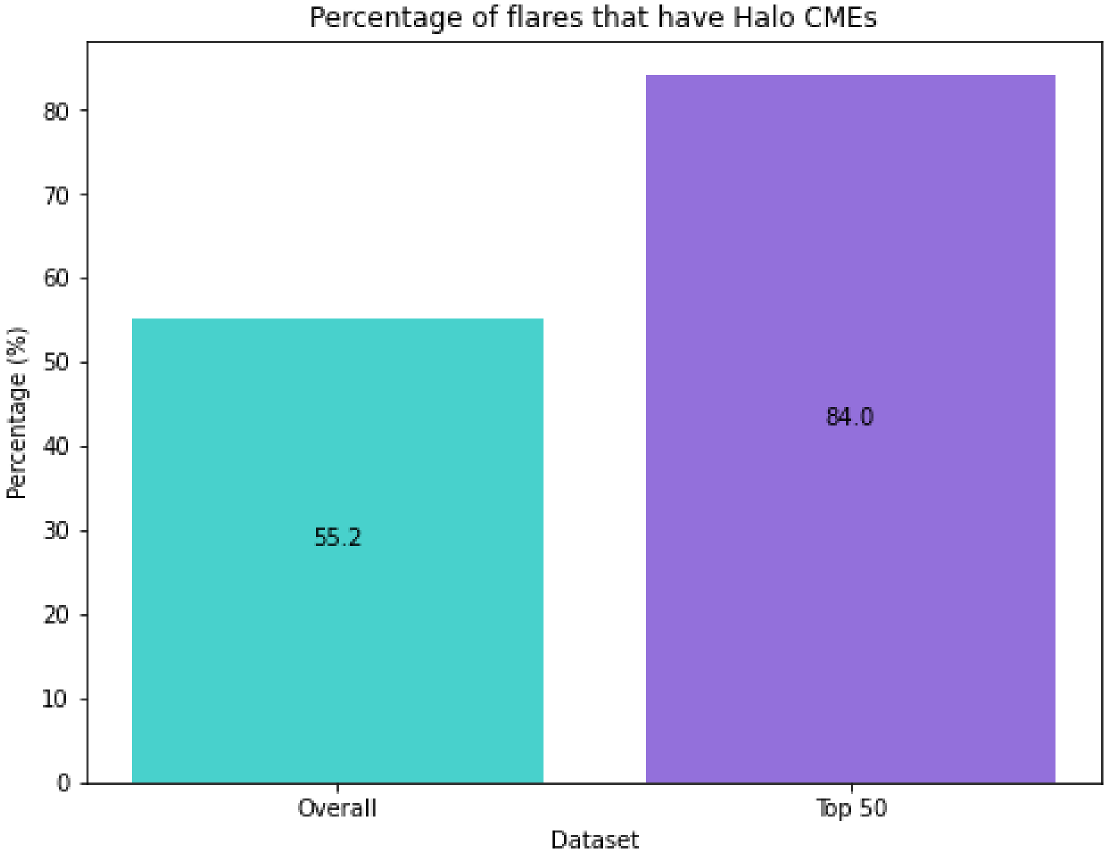
```
[8]:         start_datetime        end_datetime start_frequency end_frequency  \
     240 2003-11-04 20:00:00 2003-11-05 00:00:00           10000           200
     117 2001-04-02 22:05:00 2001-04-03 02:30:00           14000           250
     234 2003-10-29 20:55:00 2003-10-30 00:00:00           11000           500
     8   1997-11-06 12:20:00 1997-11-07 08:30:00           14000           100
     328 2006-12-05 10:50:00 2006-12-05 20:00:00           14000           250
     237 2003-11-02 17:30:00 2003-11-03 01:00:00           12000           250
     288 2005-01-20 07:15:00 2005-01-20 16:30:00           14000            25
     331 2006-12-06 19:00:00 2006-12-09 00:00:00           16000            30
     317 2005-09-09 19:45:00 2005-09-09 22:00:00           10000            50
     82  2000-07-14 10:30:00 2000-07-15 14:30:00           14000            80
     121 2001-04-06 19:35:00 2001-04-07 01:50:00           14000           230
     135 2001-08-25 16:50:00 2001-08-25 23:00:00            8000           170
     193 2002-07-23 00:50:00 2002-07-23 04:00:00           11000           400
     104 2000-11-26 17:00:00 2000-11-26 17:15:00           14000          7000
     239 2003-11-03 10:00:00 2003-11-03 12:30:00            6000           400
     286 2005-01-17 10:00:00 2005-01-17 10:35:00            6100          1500
     222 2003-05-28 01:00:00 2003-05-29 00:30:00            1000           200
     332 2006-12-13 02:45:00 2006-12-13 10:40:00           12000           150
     192 2002-07-20 21:30:00 2002-07-20 22:20:00           10000          2000
     201 2002-08-24 01:45:00 2002-08-24 03:25:00            5000           400
     238 2003-11-03 01:15:00 2003-11-03 01:25:00            3000          1500
     19  1998-05-06 08:25:00 1998-05-06 08:35:00           14000          5000
     284 2005-01-15 23:00:00 2005-01-17 00:00:00            3000            40
     142 2001-09-24 10:45:00 2001-09-25 20:00:00            7000            30

        flare_location flare_region flare_classification         cme_datetime  \
     240         S19W83        10486                 X28.  2003-11-04 19:54:00
     117         N19W72         9393                 X20.  2001-04-02 22:06:00
     234         S15W02        10486                 X10.  2003-10-29 20:54:00
     8           S18W63         8100                 X9.4  1997-11-06 12:10:00
     328         S07E68        10930                 X9.0                  NaN
     237         S14W56        10486                 X8.3  2003-11-02 17:30:00
     288         N14W61        10720                 X7.1  2005-01-20 06:54:00
     331         S05E64        10930                 X6.5                  NaN
     317         S12E67        10808                 X6.2  2005-09-09 19:48:00
     82          N22W07         9077                 X5.7  2000-07-14 10:54:00
     121         S21E31         9415                 X5.6  2001-04-06 19:30:00
     135         S17E34         9591                 X5.3  2001-08-25 16:50:00
     193         S13E72        10039                 X4.8  2002-07-23 00:42:00
     104         N18W38         9236                 X4.0  2000-11-26 17:06:00
     239         N08W77        10488                 X3.9  2003-11-03 10:06:00
     286         N15W25        10720                 X3.8  2005-01-17 09:54:00
     222         S07W20        10365                 X3.6  2003-05-28 00:50:00
     332         S06W23        10930                 X3.4  2006-12-13 02:54:00
     192         S13E90        10039                 X3.3  2002-07-20 22:06:00
     201         S02W81        10069                 X3.1  2002-08-24 01:27:00
```

```
238         N10W83        10488                 X2.7  2003-11-03 01:59:00
19          S11W65         8210                 X2.7  1998-05-06 08:29:00
284         N15W05        10720                 X2.6  2005-01-15 23:06:00
142         S16E23         9632                 X2.6  2001-09-24 10:30:00

    cme_angle cme_width cme_speed  is_halo  is_lower_bound rank
240       NaN       360      2657     True           False    1
117       261       244      2505    False           False    2
234       NaN       360      2029     True           False    6
8         NaN       360      1556     True           False    7
328       NaN       NaN       NaN    False           False    9
237       NaN       360      2598     True           False   10
288       NaN       360       882     True           False   12
331       NaN       NaN       NaN    False           False   14
317       NaN       360      2257     True           False   15
82        NaN       360      1674     True           False   17
121       NaN       360      1270     True           False   18
135       NaN       360      1433     True           False   22
193       NaN       360      2285     True           False   25
104       NaN       360       980     True           False   26
239       293       103      1420    False           False   27
286       NaN       360      2547     True           False   29
222       NaN       360      1366     True           False   33
332       NaN       360      1774     True           False   34
192       NaN       360      1941     True           False   37
201       NaN       360      1913     True           False   41
238       304        65       827    False           False   47
19        309       190      1099    False           False   48
284       NaN       360      2861     True           False   49
142       NaN       360      2402     True           False   50
```

I chose to match the NASA data with the SpaceWeatherLive data with flare classification, region, and starting date. Before matching, I noticed a pattern between the SpaceWeatherLive regions and the NASA region. For all SpaceWeatherLive regions represented with 3 or 2 digits, adding '10' and '100' in front the region numbers respectively results in a matching region number in NASA's data. I wrote a simple funcion to convert SpaceWeatherLive's region number format into NASA's and matched the two tables. My match criterion resulted in 24 total matches.

[9]:
```
### Part 2 Step 3

# Created two counters counting the number of Halos in the top 50 and the␣
 ↪entire NASA dataset
# Found the percentage of Halos in both data by dividing the two counts by␣
 ↪number of rows for each datasets and multiplying by 100.
# Used matplotlib to plot the bar graphs.

import matplotlib.pyplot as plt
```

```
sf_halo = 0
for i, row in sf50.iterrows():
  if sf50.at[i, 'is_halo'] == True:
    sf_halo += 1

nasa_halo = 0
for i, row in nasa_data.iterrows():
  if nasa_data.at[i, 'is_halo'] == True:
    nasa_halo += 1

fig = plt.figure(figsize = (8, 6))
category = ['Overall', 'Top 50']
values = [(nasa_halo/len(nasa_data)) * 100, (sf_halo/len(sf50)) * 100]
bar = plt.bar(category, values, color = ['mediumturquoise', 'mediumpurple'])
plt.title('Percentage of flares that have Halo CMEs')
plt.xlabel('Dataset')
plt.ylabel('Percentage (%)')
for rect in bar:
  height = rect.get_height()
  plt.text(rect.get_x() + rect.get_width() / 2, height / 2, f'{height:.1f}',␣
 ↪ha='center', va='bottom')

plt.show()
```



The intent of my plot is to show given the overall dataset as well as the top 50 solar flares, which data has a higher proportion of having Halo CMEs. The plot above shows a bar graph with two bars; the turquois bar represents the percentage of flares that have Halo CMEs for the overall NASA dataset, and the purple one represents percentage for the top 50 solar flares. As shown, approximately 55.2% of the solar flares in the overall NASA dataset have Halo CMEs compared to 84% in the top 50 solar flares. From the bar graph, we can see that the top 50 solar flares have a higher chance of having Halo CMEs and that there may be a correlation between solar flare classification and their likelihood of having Halo CMEs that can be further studied.