session-4

September 2, 2024

1 Session_4

[1]:
```
import pandas as pd
```

[4]:
```
dataset = pd.read_csv('hacker.csv')
```

[7]:
```
dataset.head(3)
```

[7]:
```
         id                                              title  \
0  12224879                          Interactive Dynamic Video
1  10975351  How to Use Open Source and Shut the Fuck Up at…
2  11964716  Florida DJs May Face Felony for April Fools' W…

                                                 url  num_points  \
0           http://www.interactivedynamicvideo.com/         386
1  http://hueniverse.com/2016/01/26/how-to-use-op…          39
2  http://www.thewire.com/entertainment/2013/04/f…           2

   num_comments    author       created_at
0            52  ne0phyte   8/4/2016 11:52
1            10    josep2  1/26/2016 19:30
2             1  vezycash  6/23/2016 22:20
```

[6]:
```
dataset['created_at']
```

[6]:
```
0          8/4/2016 11:52
1         1/26/2016 19:30
2         6/23/2016 22:20
3          6/17/2016 0:01
4          9/30/2015 4:12
               …
20095      8/29/2016 2:22
20096     10/6/2015 14:57
20097       1/2/2016 0:49
20098    12/15/2015 19:32
20099      5/12/2016 1:43
Name: created_at, Length: 20100, dtype: object
```

[10]:
```
type(dataset['created_at'])
```

[10]:
```
pandas.core.series.Series
```

2 to_datetime()

[13]:
```
dataset['created_at'] = pd.to_datetime(dataset['created_at'])
```

[15]:
```
dataset.head(3)
```

[15]:

	id	title	\
0	12224879	Interactive Dynamic Video	
1	10975351	How to Use Open Source and Shut the Fuck Up at…	
2	11964716	Florida DJs May Face Felony for April Fools' W…	

	url	num_points	\
0	http://www.interactivedynamicvideo.com/	386	
1	http://hueniverse.com/2016/01/26/how-to-use-op…	39	
2	http://www.thewire.com/entertainment/2013/04/f…	2	

	num_comments	author	created_at
0	52	ne0phyte	2016-08-04 11:52:00
1	10	josep2	2016-01-26 19:30:00
2	1	vezycash	2016-06-23 22:20:00

3 Skill Test - 1

Extract hours from created_at column

[16]:
```
dataset['created_at']
```

[16]:
```
0       2016-08-04 11:52:00
1       2016-01-26 19:30:00
2       2016-06-23 22:20:00
3       2016-06-17 00:01:00
4       2015-09-30 04:12:00
               …
20095   2016-08-29 02:22:00
20096   2015-10-06 14:57:00
20097   2016-01-02 00:49:00
20098   2015-12-15 19:32:00
20099   2016-05-12 01:43:00
Name: created_at, Length: 20100, dtype: datetime64[ns]
```

[20]:
```
dataset['created_at_hour'] = dataset['created_at'].dt.hour
```

[21]:
```
dataset['created_at_hour']
```

[21]:
```
0        11
1        19
2        22
3         0
4         4
         ..
20095     2
20096    14
20097     0
20098    19
20099     1
Name: created_at_hour, Length: 20100, dtype: int64
```

[25]:
```
dataset['title'] = dataset['title'].str.lower()
```

[26]:
```
dataset['title']
```

[26]:
```
0                                interactive dynamic video
1        how to use open source and shut the fuck up at…
2        florida djs may face felony for april fools' w…
3             technology ventures: from idea to enterprise
4        note by note: the making of steinway l1037 (2007)
                               …
20095    how purism avoids intels active management tec…
20096            yc application translated and broken down
20097    microkernels are slow and elvis didn't do no d…
20098                         how product hunt really works
20099    robobrowser: your friendly neighborhood web sc…
Name: title, Length: 20100, dtype: object
```

4 Skill Test - 2

- Filter records for Ask HN

[84]:
```
ask = dataset['title'].str.startswith('ask hn')
```

[85]:
```
ask
```

[85]:
```
0        False
1        False
2        False
3        False
4        False
         …
20095    False
```

```
20096    False
20097    False
20098    False
20099    False
Name: title, Length: 20100, dtype: bool
```

5 Skill Test - 3

- Filter records show hn

[42]:
```
show = dataset['title'].str.startswith('show hn')
```

[43]:
```
show
```

[43]:
```
0        False
1        False
2        False
3        False
4        False
         …
20095    False
20096    False
20097    False
20098    False
20099    False
Name: title, Length: 20100, dtype: bool
```

6 Skill Test - 4

- Filter records other than ask hn and show hn

7 Logical operators for boolean indexing in pandas

- & #and
- ~ #not
- #or

[56]:
```
dataset[~(ask) & ~(show)]
```

[56]:
```
             id                                                title  \
0      12224879                            interactive dynamic video
1      10975351  how to use open source and shut the fuck up at…
2      11964716  florida djs may face felony for april fools' w…
3      11919867       technology ventures: from idea to enterprise
4      10301696  note by note: the making of steinway l1037 (2007)
```

```
…          …                                                  …
20095  12379592  how purism avoids intels active management tec…
20096  10339284           yc application translated and broken down
20097  10824382  microkernels are slow and elvis didn't do no d…
20098  10739875                        how product hunt really works
20099  11680777  robobrowser: your friendly neighborhood web sc…

                                                    url  num_points  \
0                 http://www.interactivedynamicvideo.com/          386
1       http://hueniverse.com/2016/01/26/how-to-use-op…           39
2       http://www.thewire.com/entertainment/2013/04/f…            2
3       https://www.amazon.com/Technology-Ventures-Ent…            3
4       http://www.nytimes.com/2007/11/07/movies/07ste…            8
…                                                     …          …
20095   https://puri.sm/philosophy/how-purism-avoids-i…           10
20096   https://medium.com/@zreitano/the-yc-applicatio…            4
20097   http://blog.darknedgy.net/technology/2016/01/0…          169
20098   https://medium.com/@benjiwheeler/how-product-h…          695
20099               https://github.com/jmcarp/robobrowser          182

       num_comments          author          created_at  created_at_hour
0                52       ne0phyte 2016-08-04 11:52:00               11
1                10         josep2 2016-01-26 19:30:00               19
2                 1       vezycash 2016-06-23 22:20:00               22
3                 1        hswarna 2016-06-17 00:01:00                0
4                 2     walterbell 2015-09-30 04:12:00                4
…               …            …                …              …
20095             6  AdmiralAsshat 2016-08-29 02:22:00                2
20096             1       zreitano 2015-10-06 14:57:00               14
20097           132    vezzy-fnord 2016-01-02 00:49:00                0
20098           222          brw12 2015-12-15 19:32:00               19
20099            58      pmoriarty 2016-05-12 01:43:00                1

[17194 rows x 8 columns]
```

[48]:
```
dataset[ask].shape
```

[48]: (1744, 8)

[49]:
```
dataset[show].shape
```

[49]: (1162, 8)

[50]:
```
dataset.shape
```

[50]: (20100, 8)

[51]:
```
20100-1162-1744
```

[51]:
```
17194
```

8 Skill Test - 5

Calculate average number of comments'ask hn' posts recieve

[62]:
```
round(dataset[ask]['num_comments'].mean(), 2)
```

[62]:
```
14.04
```

9 Skill Test - 6

Calculate average number of comments'show hn' posts recieve

[63]:
```
round(dataset[show]['num_comments'].mean(), 2)
```

[63]:
```
10.32
```

10 Skill Test -7

- What is the optimal hour to post, based on your analysis

[68]:
```
groupby_hour = dataset.groupby(by = 'created_at_hour')
```

[70]:
```
#groupby_hour.get_group(7)
```

[74]:
```
groupby_hour['num_comments'].mean().sort_values(ascending=False)
```

[74]:
```
created_at_hour
14    29.144222
15    29.018639
13    27.733212
12    27.465872
11    27.118110
2     26.015123
17    25.538913
18    25.188995
9     25.080460
0     25.076040
7     24.755906
10    24.516035
19    24.361572
8     24.328720
3     23.823770
```

```
16    23.699693
5     22.715232
23    22.598972
20    22.277831
21    21.992233
4     21.891841
22    21.353143
1     21.198980
6     19.771368
Name: num_comments, dtype: float64
```

[73]:
```
groupby_hour['num_comments'].mean().max()
```

[73]:
```
29.14422241529105
```

[75]:
```
groupby_hour['num_comments'].mean().sort_values(ascending=False).head(1)
```

[75]:
```
created_at_hour
14    29.144222
Name: num_comments, dtype: float64
```

11 Skill Test - 8

- what are the top 5 recommended hours to post for 'ask hn' based on your analysis

[86]:
```
groupby_ask = dataset[ask].groupby('created_at_hour')
```

[88]:
```
groupby_ask['num_comments'].mean().sort_values(ascending=False).head()
```

[88]:
```
created_at_hour
15    38.594828
2     23.810345
20    21.525000
16    16.796296
21    16.009174
Name: num_comments, dtype: float64
```

12 Skill Test - 9

- what are the top 5 recommended hours to post for 'show hn' based on your analysis

[93]:
```
groupby_show = dataset[show].groupby('created_at_hour')
```

[94]:
```
groupby_show['num_comments'].mean().sort_values(ascending=False).head()
```

[94]:
```
created_at_hour
18    15.770492
0     15.709677
14    13.441860
23    12.416667
22    12.391304
Name: num_comments, dtype: float64
```

13 pivot table

[97]:
```
avgby_hour = dataset[ask].pivot_table(values = 'num_comments',␣
↪index='created_at_hour', aggfunc='mean')
```

[99]:
```
avgby_hour.sort_values('num_comments', ascending=False).head()
```

[99]:

created_at_hour	num_comments
15	38.594828
2	23.810345
20	21.525000
16	16.796296
21	16.009174

[ ]:
```

```