Zeru-Zhou-project7

February 27, 2022

1 Project 7 – Zeru Zhou

TA Help: NA

Collaboration: NA

- Get help from Dr. Ward's videos

1.1 Question 1

[1]:
```
import pandas as pd
import numpy as np
from pathlib import Path
```

[2]:
```
csv_files = Path('/depot/datamine/data/disney').glob('*.csv')
```

[3]:
```
data = pd.DataFrame()
for csv in csv_files:
    if csv.name == 'metadata.csv' or csv.name == 'entities.csv':
        continue
    df = pd.read_csv(csv)
    df['ride_name'] = csv.name.split('.')[0]
    data = pd.concat([data,df])
```

[15]:
```
data.head()
```

[15]:

	date	datetime	SACTMIN	SPOSTMIN	ride_name	\
0	01/01/2015	2015-01-01 08:27:58	NaN	-999.0	pirates_of_caribbean	
1	01/01/2015	2015-01-01 08:56:55	NaN	-999.0	pirates_of_caribbean	
2	01/01/2015	2015-01-01 08:57:38	NaN	-999.0	pirates_of_caribbean	
3	01/01/2015	2015-01-01 08:58:34	NaN	-999.0	pirates_of_caribbean	
4	01/01/2015	2015-01-01 09:24:55	NaN	-999.0	pirates_of_caribbean	

	code	name	short_name	land	opened_on	duration	average_wait_per_hundred
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN

[4]:
```
data = data.drop(columns = ['date'])
data.head()
```

[4]:

	datetime	SACTMIN	SPOSTMIN	ride_name
0	2015-01-01 08:27:58	NaN	-999.0	pirates_of_caribbean
1	2015-01-01 08:56:55	NaN	-999.0	pirates_of_caribbean
2	2015-01-01 08:57:38	NaN	-999.0	pirates_of_caribbean
3	2015-01-01 08:58:34	NaN	-999.0	pirates_of_caribbean
4	2015-01-01 09:24:55	NaN	-999.0	pirates_of_caribbean

[5]:
```
data['status'] = 'open'
```

[6]:
```
data.loc[data['SPOSTMIN'] == -999, "status"] = "closed"
```

[7]:
```
data.loc[data['SPOSTMIN'] == -999, "SPOSTMIN"] = np.nan
```

[8]:
```
data.head()
```

[8]:

	datetime	SACTMIN	SPOSTMIN	ride_name	status
0	2015-01-01 08:27:58	NaN	NaN	pirates_of_caribbean	closed
1	2015-01-01 08:56:55	NaN	NaN	pirates_of_caribbean	closed
2	2015-01-01 08:57:38	NaN	NaN	pirates_of_caribbean	closed
3	2015-01-01 08:58:34	NaN	NaN	pirates_of_caribbean	closed
4	2015-01-01 09:24:55	NaN	NaN	pirates_of_caribbean	closed

[9]:
```
data['datetime'] = pd.to_datetime(data['datetime'])
```

[10]:
```
data['ride_name'] = data['ride_name'].astype("category")
data['status'] = data['status'].astype("category")
```

[40]:
```
print(data.dtypes)
```

```
datetime     datetime64[ns]
SACTMIN             float64
SPOSTMIN            float64
ride_name          category
status             category
dtype: object
```

[11]:
```
data.reset_index(drop = True, inplace = True)
```

[12]:
```
data.tail()
```

[12]:

	datetime	SACTMIN	SPOSTMIN	ride_name	status
3443440	2021-12-28 20:32:16	NaN	75.0	slinky_dog	open
3443441	2021-12-28 20:40:57	NaN	75.0	slinky_dog	open
3443442	2021-12-28 20:46:27	NaN	70.0	slinky_dog	open

```
3443443 2021-12-28 20:48:31      NaN      65.0  slinky_dog   open
3443444 2021-12-28 21:00:21      NaN      65.0  slinky_dog   open
```

As code above, data is prepared.

1.2 Question 2

[43]:
```
data.groupby("ride_name").mean().sort_values('SACTMIN', ascending = False)
```

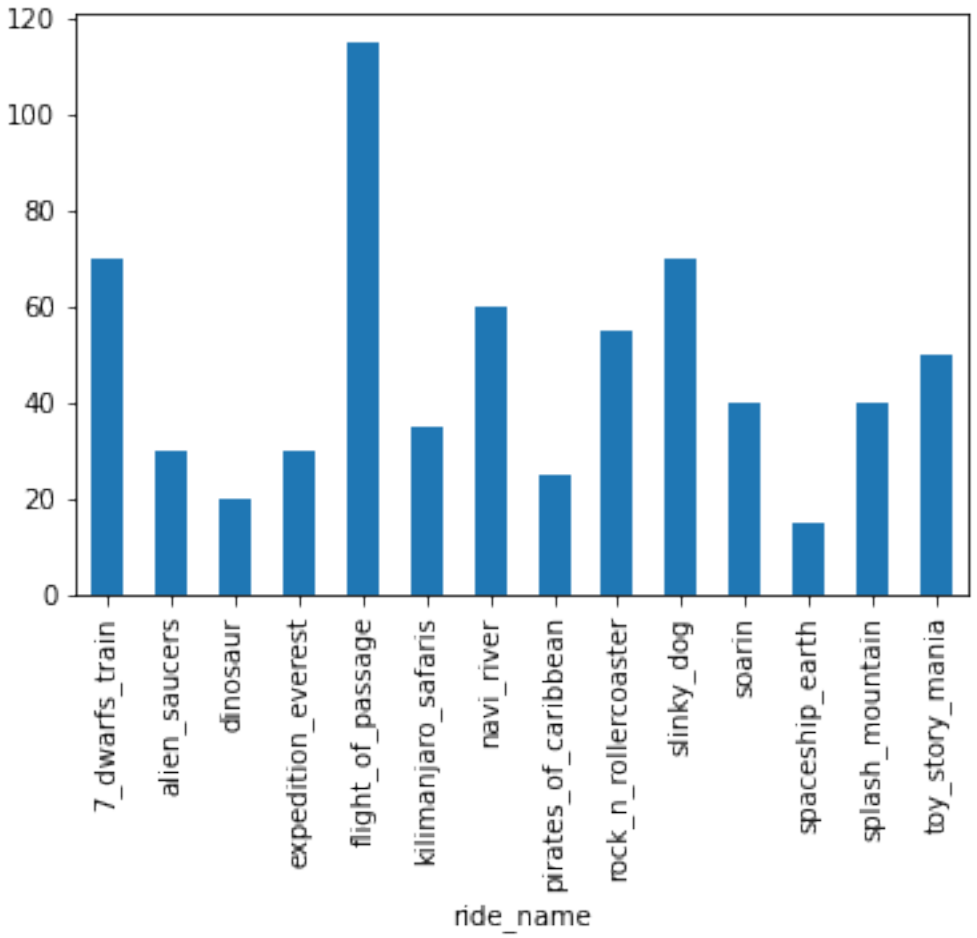
[43]:
```
                         SACTMIN    SPOSTMIN
ride_name
flight_of_passage     72.917854  114.806860
slinky_dog            40.657119   72.703003
toy_story_mania       35.066966   54.264905
rock_n_rollercoaster  29.267599   58.994605
navi_river            29.046130   62.511237
splash_mountain       25.430836   43.731363
soarin                24.963390   45.654540
7_dwarfs_train        23.991880   76.962455
kilimanjaro_safaris   22.216697   39.959134
alien_saucers         22.068624   29.863351
dinosaur              20.197075   27.399704
pirates_of_caribbean  18.167141   28.524095
expedition_everest    15.500858   32.429036
spaceship_earth        9.757474   18.790590
```

As above, mean is calculated and sorted.

1.3 Question 3

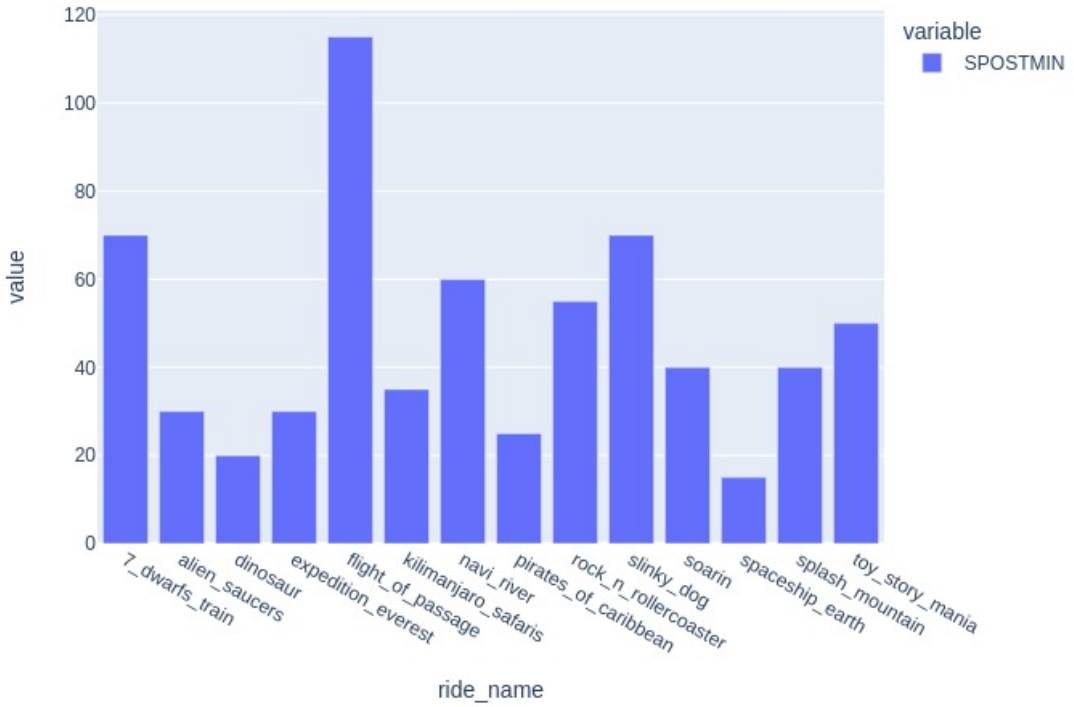
[45]:
```
dat = data.groupby("ride_name")['SPOSTMIN'].median()
dat.plot.bar()
```

[45]:
```
<AxesSubplot:xlabel='ride_name'>
```



[13]:
```
import plotly.express as px
```

[48]:
```
fig = px.bar(data.groupby("ride_name")['SPOSTMIN'].median())
fig.show(renderer = 'jpg')
```



As above, two bar plot are created.

1.4 Question 4

[14]:
```
def min_to_hr(minute):
    return minute/60.0
```

[15]:
```
data.groupby('ride_name').median().apply(min_to_hr).query("SPOSTMIN >= 1")
```

[15]:

ride_name	SACTMIN	SPOSTMIN
7_dwarfs_train	0.516667	1.166667
flight_of_passage	0.883333	1.916667
navi_river	0.400000	1.000000
slinky_dog	0.616667	1.166667

Apply and query commands are used.

1.5 Question 5

[18]:
```
data.assign(mean_wait_time_act = lambda x: x.groupby('ride_name')['SACTMIN'].
↪transform('mean') )
```

[18]:
```
                  datetime  SACTMIN  SPOSTMIN             ride_name  status  \
0       2015-01-01 08:27:58      NaN       NaN  pirates_of_caribbean  closed
1       2015-01-01 08:56:55      NaN       NaN  pirates_of_caribbean  closed
2       2015-01-01 08:57:38      NaN       NaN  pirates_of_caribbean  closed
3       2015-01-01 08:58:34      NaN       NaN  pirates_of_caribbean  closed
4       2015-01-01 09:24:55      NaN       NaN  pirates_of_caribbean  closed
…                       …      …         …                    …      …
3443440 2021-12-28 20:32:16      NaN      75.0            slinky_dog    open
3443441 2021-12-28 20:40:57      NaN      75.0            slinky_dog    open
3443442 2021-12-28 20:46:27      NaN      70.0            slinky_dog    open
3443443 2021-12-28 20:48:31      NaN      65.0            slinky_dog    open
3443444 2021-12-28 21:00:21      NaN      65.0            slinky_dog    open

         mean_wait_time_act
0                 18.167141
1                 18.167141
2                 18.167141
3                 18.167141
4                 18.167141
…                       …
3443440           40.657119
3443441           40.657119
3443442           40.657119
3443443           40.657119
3443444           40.657119

[3443445 rows x 6 columns]
```

[20]:
```
data.assign(mean_wait_time_post = lambda x: x.groupby('ride_name')['SPOSTMIN'].
↪transform('mean'))
```

[20]:
```
                  datetime  SACTMIN  SPOSTMIN             ride_name  status  \
0       2015-01-01 08:27:58      NaN       NaN  pirates_of_caribbean  closed
1       2015-01-01 08:56:55      NaN       NaN  pirates_of_caribbean  closed
2       2015-01-01 08:57:38      NaN       NaN  pirates_of_caribbean  closed
3       2015-01-01 08:58:34      NaN       NaN  pirates_of_caribbean  closed
4       2015-01-01 09:24:55      NaN       NaN  pirates_of_caribbean  closed
…                       …      …         …                    …      …
3443440 2021-12-28 20:32:16      NaN      75.0            slinky_dog    open
3443441 2021-12-28 20:40:57      NaN      75.0            slinky_dog    open
3443442 2021-12-28 20:46:27      NaN      70.0            slinky_dog    open
3443443 2021-12-28 20:48:31      NaN      65.0            slinky_dog    open
```

```
3443444 2021-12-28 21:00:21      NaN      65.0            slinky_dog    open

         mean_wait_time_post
0                  28.524095
1                  28.524095
2                  28.524095
3                  28.524095
4                  28.524095
…                        …
3443440            72.703003
3443441            72.703003
3443442            72.703003
3443443            72.703003
3443444            72.703003

[3443445 rows x 6 columns]
```

[21]:
```
data.assign(median_wait_time_act = lambda x: x.groupby('ride_name')['SACTMIN'].
↪transform('median'))
```

[21]:
```
                  datetime  SACTMIN  SPOSTMIN             ride_name  status  \
0       2015-01-01 08:27:58      NaN       NaN  pirates_of_caribbean  closed
1       2015-01-01 08:56:55      NaN       NaN  pirates_of_caribbean  closed
2       2015-01-01 08:57:38      NaN       NaN  pirates_of_caribbean  closed
3       2015-01-01 08:58:34      NaN       NaN  pirates_of_caribbean  closed
4       2015-01-01 09:24:55      NaN       NaN  pirates_of_caribbean  closed
…                       …      …        …                    …     …
3443440 2021-12-28 20:32:16      NaN      75.0            slinky_dog    open
3443441 2021-12-28 20:40:57      NaN      75.0            slinky_dog    open
3443442 2021-12-28 20:46:27      NaN      70.0            slinky_dog    open
3443443 2021-12-28 20:48:31      NaN      65.0            slinky_dog    open
3443444 2021-12-28 21:00:21      NaN      65.0            slinky_dog    open

         median_wait_time_act
0                        16.0
1                        16.0
2                        16.0
3                        16.0
4                        16.0
…                        …
3443440                  37.0
3443441                  37.0
3443442                  37.0
3443443                  37.0
3443444                  37.0

[3443445 rows x 6 columns]
```

[22]:
```
data.assign(median_wait_time_post = lambda x: x.
↪groupby('ride_name')['SPOSTMIN'].transform('median'))
```

[22]:

```
                   datetime  SACTMIN  SPOSTMIN             ride_name  status  \
0       2015-01-01 08:27:58      NaN       NaN  pirates_of_caribbean  closed
1       2015-01-01 08:56:55      NaN       NaN  pirates_of_caribbean  closed
2       2015-01-01 08:57:38      NaN       NaN  pirates_of_caribbean  closed
3       2015-01-01 08:58:34      NaN       NaN  pirates_of_caribbean  closed
4       2015-01-01 09:24:55      NaN       NaN  pirates_of_caribbean  closed
…                       …      …        …                     …      …
3443440 2021-12-28 20:32:16      NaN      75.0            slinky_dog    open
3443441 2021-12-28 20:40:57      NaN      75.0            slinky_dog    open
3443442 2021-12-28 20:46:27      NaN      70.0            slinky_dog    open
3443443 2021-12-28 20:48:31      NaN      65.0            slinky_dog    open
3443444 2021-12-28 21:00:21      NaN      65.0            slinky_dog    open

         median_wait_time_post
0                         25.0
1                         25.0
2                         25.0
3                         25.0
4                         25.0
…                         …
3443440                   70.0
3443441                   70.0
3443442                   70.0
3443443                   70.0
3443444                   70.0

[3443445 rows x 6 columns]
```

Four new columns are listed above.

1.6 Pledge

By submitting this work I hereby pledge that this is my own, personal work. I've acknowledged in the designated place at the top of this file all sources that I used to complete said work, including but not limited to: online resources, books, and electronic communications. I've noted all collaboration with fellow students and/or TA's. I did not copy or plagiarize another's work.

> As a Boilermaker pursuing academic excellence, I pledge to be honest and true in all that I do. Accountable together – We are Purdue.