Pandas - Assignment 01_solution

June 22, 2020

1 Assignment 01: Evaluate the FAA Dataset

The comments/sections provided are your cues to perform the assignment. You don't need to limit yourself to the number of rows/cells provided. You can add additional rows in each section to add more lines of code.

If at any point in time you need help on solving this assignment, view our demo video to understand the different steps of the code.

Happy coding!

1: VIew and import the dataset

[3]:
```
#Import necessary libraries
import pandas as pd
```

[4]:
```
#Import the FAA (Federal Aviation Authority) dataset
df_fa_dataset = pd.read_csv("faa_ai_prelim.csv")
```

2: View and understand the dataset

[6]:
```
#View the dataset shape
df_fa_dataset.shape
```

[6]:
```
(83, 42)
```

[7]:
```
#View the first five observations
df_fa_dataset.head()
```

[7]:
```
  UPDATED ENTRY_DATE EVENT_LCL_DATE EVENT_LCL_TIME LOC_CITY_NAME  \
0      No  19-FEB-16      19-FEB-16      00:45:00Z    MARSHVILLE
1      No  19-FEB-16      18-FEB-16      23:55:00Z     TAVERNIER
2      No  19-FEB-16      18-FEB-16      22:14:00Z       TRENTON
3      No  19-FEB-16      18-FEB-16      17:10:00Z     ASHEVILLE
4      No  19-FEB-16      18-FEB-16      00:26:00Z     TALKEETNA

   LOC_STATE_NAME LOC_CNTRY_NAME  \
```

```
0  North Carolina            NaN
1         Florida            NaN
2      New Jersey            NaN
3  North Carolina            NaN
4          Alaska            NaN

                                          RMK_TEXT EVENT_TYPE_DESC  \
0  AIRCRAFT CRASHED INTO TREES, THE 1 PERSON ON B…        Accident
1  AIRCRAFT ON LANDING WENT OFF THE END OF THE RU…        Incident
2  AIRCRAFT ON FINAL SUSTAINED A BIRD STRIKE, LAN…        Incident
3  AIRCRAFT ON LANDING, GEAR COLLAPSED, ASHEVILLE…        Incident
4  AIRCRAFT ON LANDING, NOSE GEAR COLLAPSED, TALK…        Incident

                  FSDO_DESC  … PAX_INJ_NONE PAX_INJ_MINOR PAX_INJ_SERIOUS  \
0     FAA Charlotte FSDO-68  …          NaN           NaN              NaN
1         FAA Miami FSDO-19  …          NaN           NaN              NaN
2  FAA Philadelphia FSDO-17  …          NaN           NaN              NaN
3     FAA Charlotte FSDO-68  …          NaN           NaN              NaN
4     FAA Anchorage FSDO-03  …          NaN           1.0              NaN

  PAX_INJ_FATAL PAX_INJ_UNK  GRND_INJ_NONE GRND_INJ_MINOR GRND_INJ_SERIOUS  \
0           NaN         NaN            NaN            NaN               NaN
1           NaN         NaN            NaN            NaN               NaN
2           NaN         NaN            NaN            NaN               NaN
3           NaN         NaN            NaN            NaN               NaN
4           NaN         NaN            NaN            NaN               NaN

  GRND_INJ_FATAL  GRND_INJ_UNK
0            NaN           NaN
1            NaN           NaN
2            NaN           NaN
3            NaN           NaN
4            NaN           NaN

[5 rows x 42 columns]
```

[9]:
```
#View all the columns present in the dataset
df_fa_dataset.columns
```

[9]:
```
Index(['UPDATED', 'ENTRY_DATE', 'EVENT_LCL_DATE', 'EVENT_LCL_TIME',
       'LOC_CITY_NAME', 'LOC_STATE_NAME', 'LOC_CNTRY_NAME', 'RMK_TEXT',
       'EVENT_TYPE_DESC', 'FSDO_DESC', 'REGIST_NBR', 'FLT_NBR', 'ACFT_OPRTR',
       'ACFT_MAKE_NAME', 'ACFT_MODEL_NAME', 'ACFT_MISSING_FLAG',
       'ACFT_DMG_DESC', 'FLT_ACTIVITY', 'FLT_PHASE', 'FAR_PART', 'MAX_INJ_LVL',
       'FATAL_FLAG', 'FLT_CRW_INJ_NONE', 'FLT_CRW_INJ_MINOR',
       'FLT_CRW_INJ_SERIOUS', 'FLT_CRW_INJ_FATAL', 'FLT_CRW_INJ_UNK',
       'CBN_CRW_INJ_NONE', 'CBN_CRW_INJ_MINOR', 'CBN_CRW_INJ_SERIOUS',
```

```
       'CBN_CRW_INJ_FATAL', 'CBN_CRW_INJ_UNK', 'PAX_INJ_NONE', 'PAX_INJ_MINOR',
       'PAX_INJ_SERIOUS', 'PAX_INJ_FATAL', 'PAX_INJ_UNK', 'GRND_INJ_NONE',
       'GRND_INJ_MINOR', 'GRND_INJ_SERIOUS', 'GRND_INJ_FATAL', 'GRND_INJ_UNK'],
      dtype='object')
```

3: Extract the following attributes from the dataset:

1. Aircraft make name
2. State name
3. Aircraft model name
4. Text information
5. Flight phase
6. Event description type
7. Fatal flag

[11]:
```
#Create a new dataframe with only the required columns
df_analyze_dataset=df_fa_dataset[['ACFT_MAKE_NAME','LOC_STATE_NAME','ACFT_MODEL_NAME','RMK_TEX
```

[12]:
```
#View the type of the object
type(df_analyze_dataset)
```

[12]:
```
pandas.core.frame.DataFrame
```

[13]:
```
#Check if the dataframe contains all the required attributes
df_analyze_dataset.head()
```

[13]:

	ACFT_MAKE_NAME	LOC_STATE_NAME	ACFT_MODEL_NAME	\
0	BEECH	North Carolina	36	
1	VANS	Florida	RV7	
2	CESSNA	New Jersey	172	
3	LANCAIR	North Carolina	235	
4	CESSNA	Alaska	172	

	RMK_TEXT	FLT_PHASE	\
0	AIRCRAFT CRASHED INTO TREES, THE 1 PERSON ON B…	UNKNOWN (UNK)	
1	AIRCRAFT ON LANDING WENT OFF THE END OF THE RU…	LANDING (LDG)	
2	AIRCRAFT ON FINAL SUSTAINED A BIRD STRIKE, LAN…	APPROACH (APR)	
3	AIRCRAFT ON LANDING, GEAR COLLAPSED, ASHEVILLE…	LANDING (LDG)	
4	AIRCRAFT ON LANDING, NOSE GEAR COLLAPSED, TALK…	LANDING (LDG)	

	EVENT_TYPE_DESC	FATAL_FLAG
0	Accident	Yes
1	Incident	NaN
2	Incident	NaN
3	Incident	NaN
4	Incident	NaN

4. Clean the dataset and replace the fatal flag NaN with "No"

[15]:
```
#Replace all Fatal Flag missing values with the required output
df_analyze_dataset['FATAL_FLAG'].fillna(value='No', inplace=True)
```

```
C:\Users\Naman\anaconda3\lib\site-packages\pandas\core\generic.py:6245:
SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  self._update_inplace(new_data)
```

[16]:
```
#Verify if the missing values are replaced
df_analyze_dataset.head()
```

[16]:

	ACFT_MAKE_NAME	LOC_STATE_NAME	ACFT_MODEL_NAME	RMK_TEXT	FLT_PHASE	EVENT_TYPE_DESC	FATAL_FLAG
0	BEECH	North Carolina	36	AIRCRAFT CRASHED INTO TREES, THE 1 PERSON ON B…	UNKNOWN (UNK)	Accident	Yes
1	VANS	Florida	RV7	AIRCRAFT ON LANDING WENT OFF THE END OF THE RU…	LANDING (LDG)	Incident	No
2	CESSNA	New Jersey	172	AIRCRAFT ON FINAL SUSTAINED A BIRD STRIKE, LAN…	APPROACH (APR)	Incident	No
3	LANCAIR	North Carolina	235	AIRCRAFT ON LANDING, GEAR COLLAPSED, ASHEVILLE…	LANDING (LDG)	Incident	No
4	CESSNA	Alaska	172	AIRCRAFT ON LANDING, NOSE GEAR COLLAPSED, TALK…	LANDING (LDG)	Incident	No

[18]:
```
#Check the number of observations
df_analyze_dataset.shape
```

[18]: (83, 7)

5. Remove all the observations where aircraft names are not available

[19]:
```
#Drop the unwanted values/observations from the dataset
df_final_dataset = df_analyze_dataset.dropna(subset=['ACFT_MAKE_NAME'])
```

6. Find the aircraft types and their occurrences in the dataset

[21]:
```
#Check the number of observations now to compare it with the original dataset␣
↪and see how many values have been dropped
df_final_dataset.shape
```

[21]:
```
(78, 7)
```

[22]:
```
#Group the dataset by aircraft name
aircraft_type = df_final_dataset.groupby('ACFT_MAKE_NAME')
```

[23]:
```
#View the number of times each aircraft type appears in the dataset (Hint: use␣
↪the size() method)
aircraft_type.size()
```

[23]:
```
ACFT_MAKE_NAME
AERO COMMANDER              1
AERONCA                     1
AEROSTAR INTERNATIONAL      1
AIRBUS                      1
BEECH                       9
BELL                        2
BOEING                      3
CESSNA                     23
CHAMPION                    2
CHRISTEN                    1
CONSOLIDATED VULTEE         1
EMBRAER                     1
ENSTROM                     1
FAIRCHILD                   1
FLIGHT DESIGN               1
GLOBE                       1
GREAT LAKES                 1
GRUMMAN                     1
GULFSTREAM                  1
HUGHES                      1
LANCAIR                     2
MAULE                       1
MOONEY                      4
NORTH AMERICAN              1
PIPER                      10
PITTS                       1
SAAB                        1
SABRELINER                  1
SOCATA                      2
VANS                        1
dtype: int64
```

7: Display the observations where fatal flag is "Yes"

[24]:
```
#Group the dataset by fatal flag
fatalAccidents = df_final_dataset.groupby('FATAL_FLAG')
```

[25]:
```
#View the total number of fatal and non-fatal accidents
fatalAccidents.size()
```

[25]:
```
FATAL_FLAG
No     71
Yes     7
dtype: int64
```

[27]:
```
#Create a new dataframe to view only the fatal accidents (Fatal Flag values =
↪Yes)
accident_with_fatality = fatalAccidents.get_group('Yes')
accident_with_fatality
```

[27]:

	ACFT_MAKE_NAME	LOC_STATE_NAME	ACFT_MODEL_NAME	RMK_TEXT	FLT_PHASE	EVENT_TYPE_DESC	FATAL_FLAG
0	BEECH	North Carolina	36	AIRCRAFT CRASHED INTO TREES, THE 1 PERSON ON B…	UNKNOWN (UNK)	Accident	Yes
53	PIPER	Florida	PA28	AIRCRAFT CRASHED UNDER UNKNOWN CIRCUMSTANCES. …	UNKNOWN (UNK)	Accident	Yes
55	FLIGHT DESIGN	California	CTLS	AIRCRAFT CRASHED UNDER UNKNOWN CIRCUMSTANCES A…	UNKNOWN (UNK)	Accident	Yes
79	NORTH AMERICAN	Arizona	F51	AIRCRAFT CRASHED UNDER UNKNOWN CIRCUMSTANCES, …	UNKNOWN (UNK)	Accident	Yes
80	CHAMPION	California	8KCAB	N9872R, BEECH M35 AIRCRAFT, AND N5057G, BELLAN…	UNKNOWN (UNK)	Accident	Yes
81	BEECH	California	35	N9872R, BEECH M35 AIRCRAFT, AND N5057G, BELLAN…	UNKNOWN (UNK)	Accident	Yes
82	CESSNA	Alabama	182	N784CP AIRCRAFT CRASHED INTO A WOODED AREA NEA…	UNKNOWN (UNK)	Accident	Yes