Day3GroupProject

April 6, 2022

[38]:
```
# Download and load the dataset into Python using .read_csv().
import pandas as pd
file_url = 'https://raw.githubusercontent.com/fenago/MLEssentials/main/datasets/
 ↪Speed_Dating_Data.csv'
df = pd.read_csv(file_url)
```

[39]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
```

[40]:
```
# Print out the dimensions of the DataFrame using .shape.
# 8378 row, 195 columns
df.shape
```

[40]:
```
(8378, 195)
```

[41]:
```
# Check for duplicate rows by using .duplicated() and .sum() on all the columns.
df.duplicated().sum()
```

[41]:
```
0
```

[42]:
```
# Check for duplicate rows by using .duplicated() and .sum() for the identifier␣
 ↪columns (iid, id, partner, and pid).
df.loc[df.duplicated().sum(), ['iid', 'id', 'partner', 'pid']]
```

[42]:
```
iid         1
id          1
partner     1
pid        11
Name: 0, dtype: object
```

[43]:
```
#Check for unexpected values for the following numerical variables:
#'imprace', 'imprelig', 'sports', 'tvsports', 'exercise', 'dining', 'museums',␣
 ↪'art', 'hiking', 'gaming', 'clubbing', 'reading', 'tv', 'theater', 'movies',␣
 ↪'concerts', 'music', 'shopping', and 'yoga'.
```

```
df[['imprace', 'imprelig', 'sports', 'tvsports', 'exercise', 'dining',␣
 ↪'museums', 'art', 'hiking', 'gaming', 'clubbing', 'reading', 'tv',␣
 ↪'theater', 'movies', 'concerts', 'music', 'shopping','yoga']]
# When one col has a missing value, all seem to have a missing value
```

[43]:

	imprace	imprelig	sports	tvsports	exercise	dining	museums	art	\
0	2.0	4.0	9.0	2.0	8.0	9.0	1.0	1.0	
1	2.0	4.0	9.0	2.0	8.0	9.0	1.0	1.0	
2	2.0	4.0	9.0	2.0	8.0	9.0	1.0	1.0	
3	2.0	4.0	9.0	2.0	8.0	9.0	1.0	1.0	
4	2.0	4.0	9.0	2.0	8.0	9.0	1.0	1.0	
…	…	…	…	…	…	…	…	…	
8373	1.0	1.0	8.0	2.0	5.0	10.0	10.0	10.0	
8374	1.0	1.0	8.0	2.0	5.0	10.0	10.0	10.0	
8375	1.0	1.0	8.0	2.0	5.0	10.0	10.0	10.0	
8376	1.0	1.0	8.0	2.0	5.0	10.0	10.0	10.0	
8377	1.0	1.0	8.0	2.0	5.0	10.0	10.0	10.0	

	hiking	gaming	clubbing	reading	tv	theater	movies	concerts	\
0	5.0	1.0	5.0	6.0	9.0	1.0	10.0	10.0	
1	5.0	1.0	5.0	6.0	9.0	1.0	10.0	10.0	
2	5.0	1.0	5.0	6.0	9.0	1.0	10.0	10.0	
3	5.0	1.0	5.0	6.0	9.0	1.0	10.0	10.0	
4	5.0	1.0	5.0	6.0	9.0	1.0	10.0	10.0	
…	…	…	…	…	…	…	…	…	
8373	7.0	1.0	9.0	8.0	3.0	7.0	9.0	10.0	
8374	7.0	1.0	9.0	8.0	3.0	7.0	9.0	10.0	
8375	7.0	1.0	9.0	8.0	3.0	7.0	9.0	10.0	
8376	7.0	1.0	9.0	8.0	3.0	7.0	9.0	10.0	
8377	7.0	1.0	9.0	8.0	3.0	7.0	9.0	10.0	

	music	shopping	yoga
0	9.0	8.0	1.0
1	9.0	8.0	1.0
2	9.0	8.0	1.0
3	9.0	8.0	1.0
4	9.0	8.0	1.0
…	…	…	…
8373	10.0	7.0	3.0
8374	10.0	7.0	3.0
8375	10.0	7.0	3.0
8376	10.0	7.0	3.0
8377	10.0	7.0	3.0

[8378 rows x 19 columns]

[44]:
```
# Dropped all the empty values
# Replace the identified incorrect values.
df2 = df.dropna(subset=['imprace', 'imprelig', 'sports', 'tvsports',␣
↪'exercise', 'dining', 'museums', 'art', 'hiking', 'gaming', 'clubbing',␣
↪'reading', 'tv', 'theater', 'movies', 'concerts', 'music',␣
↪'shopping','yoga'])
```

[45]:
```
# Data Types
# Check the data type of the different columns using .dtypes.
df2.dtypes
```

[45]:
```
iid           int64
id          float64
gender        int64
idg           int64
condtn        int64
              ...
attr5_3     float64
sinc5_3     float64
intel5_3    float64
fun5_3      float64
amb5_3      float64
Length: 195, dtype: object
```

[52]:
```
# Change the data types to categorical for the columns that don't contain␣
↪numerical values using .astype().

obj_df = df2.select_dtypes(include='object')
obj_cols = obj_df.columns

for col_name in obj_cols:
    df2[col_name] = df2[col_name].astype('category')
```

[47]:
```
df2.info()
#object values have been changed to categorical
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 8299 entries, 0 to 8377
Columns: 195 entries, iid to amb5_3
dtypes: category(4), float64(178), int64(13)
memory usage: 12.3 MB
```

[48]:
```
#Check for any missing values using .isna() and .sum() for each numerical␣
↪variable.

df2.isna().sum()
```

[48]:
```
iid             0
id              1
gender          0
idg             0
condtn          0
               …
attr5_3      6283
sinc5_3      6283
intel5_3     6283
fun5_3       6283
amb5_3       6283
Length: 195, dtype: int64
```

[49]:
```
# Replace the missing values for each numerical variable with their␣
↪corresponding mean or median values using .fillna(), .mean(), and .median().
↪NoteThe dataset for this activity can be found in this courses GitHub␣
↪repository: https://raw.githubusercontent.com/fenago/MLEssentials/main/
↪datasets/Speed_Dating_Data.csv
num_df = df2.select_dtypes(include=['int64', 'float64'])
num_cols = num_df.columns
num_cols
```

[49]:
```
Index(['iid', 'id', 'gender', 'idg', 'condtn', 'wave', 'round', 'position',
       'positin1', 'order',
       …
       'attr3_3', 'sinc3_3', 'intel3_3', 'fun3_3', 'amb3_3', 'attr5_3',
       'sinc5_3', 'intel5_3', 'fun5_3', 'amb5_3'],
      dtype='object', length=191)
```

[51]:
```
for col_name in num_cols:
    avg = df2[col_name].mean()
    df2[col_name].fillna(avg, inplace=True)
```

[54]:
```
df2[num_cols].isna().sum().max()
```

[54]:
```
0
```

[55]:
```
# Complete a Univariate / Bivariate analysis and correlation matrix to document␣
↪insights that you discover in the dataset.  This can be a spreadshot, word␣
↪doc, pdf, or zip file.
```

[ ]:

[ ]: