assignment3

May 11, 2022

0.0.1 Descriptive Statistics - Measures of Central Tendency and variability

Perform the following operations on any open source dataset (e.g., data.csv) - Provide summary statistics (mean, median, minimum, maximum, standard deviation) for a dataset (age, income etc.) with numeric variables grouped by one of the qualitative (categorical) variable. For example, if your categorical variable is age groups and quantitative variable is income, then provide summary statistics of income grouped by the age groups. Create a list that contains a numeric value for each response to the categorical variable. - Write a Python program to display some basic statistical details like percentile, mean, standard deviation etc. of the species of 'Iris-setosa', 'Iris-versicolor' and 'Iris-versicolor' of iris.csv dataset.

[6]:

```
import pandas as pd
#Dataset CSV
url = "eduPerform.csv"
df = pd.read_csv(url)
df.head(10)
```

[6]:

```
  gender NationalITy PlaceofBirth       StageID GradeID SectionID Topic  \
0    NaN          KW       KuwaIT    lowerlevel    G-04         A    IT
1      M          KW          NaN    lowerlevel    G-04         A   NaN
2      M          KW       KuwaIT           NaN    G-04         A    IT
3      M          KW       KuwaIT    lowerlevel    G-04         A    IT
4    NaN          KW       KuwaIT    lowerlevel    G-04         A    IT
5      F          KW       KuwaIT    lowerlevel    G-04         A    IT
6      M          KW       KuwaIT  MiddleSchool    G-07         A   NaN
7      M          KW          NaN  MiddleSchool    G-07         A  Math
8      F          KW       KuwaIT  MiddleSchool    G-07         A  Math
9      F          KW       KuwaIT  MiddleSchool    G-07         B    IT

  Semester Relation  cns  dsa oops os
0        F   Father  NaN 16.0    2 20
1        F   Father 20.0 20.0    3 25
2        F   Father 10.0  7.0    0 30
3        F   Father  NaN 25.0    5 35
4        F   Father 40.0 50.0   12 50
5        F   Father 42.0 30.0   13 70
6        F   Father 35.0 12.0    0 17
7        F      NaN  NaN  NaN   15 22
8        F   Father 12.0 21.0   16 50
```

```
9        F   Father   NaN  80.0    25  70
```

[10]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 28 entries, 0 to 27
Data columns (total 13 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   gender        22 non-null     object
 1   NationalITy   27 non-null     object
 2   PlaceofBirth  23 non-null     object
 3   StageID       26 non-null     object
 4   GradeID       27 non-null     object
 5   SectionID     28 non-null     object
 6   Topic         24 non-null     object
 7   Semester      28 non-null     object
 8   Relation      26 non-null     object
 9   cns           21 non-null     float64
 10  dsa           27 non-null     float64
 11  oops          28 non-null     int64
 12  os            28 non-null     int64
dtypes: float64(2), int64(2), object(9)
memory usage: 3.0+ KB
```

[11]:
```
df.max(numeric_only=True)
```

[11]:
```
cns     70.0
dsa     88.0
oops    44.0
os      99.0
dtype: float64
```

[13]:
```
#maximum for particular value in a dataset
print(df['os'].max())
```

```
99
```

[15]:
```
#min for all value in a dataset
df.min(numeric_only=True)
```

[15]:
```
cns      0.0
dsa      0.0
oops     0.0
os      11.0
dtype: float64
```

[17]:

```
#mean for all value in a dataset
print(df.mean(numeric_only=True))
```

```
cns     25.571429
dsa     26.814815
oops    16.428571
os      53.392857
dtype: float64
```

[21]:

```
#median for all value in a dataset
df.median(numeric_only=True)
```

[21]:

```
cns     20.0
dsa     19.0
oops    14.0
os      50.0
dtype: float64
```

[20]:

```
#mode for all value in a dataset
print(df.mode())
```

```
  gender NationalITy PlaceofBirth       StageID GradeID SectionID Topic  \
0      M          KW       KuwaIT  MiddleSchool    G-07         A    IT
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
5    NaN         NaN          NaN           NaN     NaN       NaN   NaN

  Semester Relation   cns   dsa  oops    os
0        F   Father  10.0   7.0   0.0  50.0
1      NaN      NaN  19.0  12.0   2.0  70.0
2      NaN      NaN  20.0  15.0  12.0  80.0
3      NaN      NaN   NaN  21.0   NaN  90.0
4      NaN      NaN   NaN  30.0   NaN   NaN
5      NaN      NaN   NaN  50.0   NaN   NaN
```

[22]:

```
#Standard deviation for all value in a dataset
df.std(numeric_only=True)
```

[22]:

```
cns     20.028908
dsa     24.334270
oops    13.658340
os      28.342272
dtype: float64
```

[23]:

```
#Variance for all value in a dataset
df.var(numeric_only=True)
```

[23]:

```
cns     401.157143
dsa     592.156695
oops    186.550265
os      803.284392
dtype: float64
```

[24]:

```
#function that prints the summary statistic of the numerical variables
df.describe()
```

[24]:

```
             cns        dsa       oops         os
count  21.000000  27.000000  28.000000  28.000000
mean   25.571429  26.814815  16.428571  53.392857
std    20.028908  24.334270  13.658340  28.342272
min     0.000000   0.000000   0.000000  11.000000
25%    10.000000  12.000000   3.000000  28.750000
50%    20.000000  19.000000  14.000000  50.000000
75%    36.000000  35.000000  26.250000  80.000000
max    70.000000  88.000000  44.000000  99.000000
```

[34]:

```
url = "eduPerform.csv"


df = pd.read_csv(url)
#Grouping and perform count over each group
df =  df.groupby('gender')['gender'].count()
print(df)
```

```
gender
F     8
M    14
Name: gender, dtype: int64
```

[33]:

```
url = "eduPerform.csv"


df = pd.read_csv(url)
#Grouping and perform sum over each group
df =  df.groupby('Topic')['Topic'].count()
print(df)
```

```
Topic
Arabic     1
IT        17
Math       6
Name: Topic, dtype: int64
```

[36]:

```python
df = pd.read_csv(url)
#Group by two keys and then summarize each group
df =  df.groupby(['gender','GradeID'],as_index=False).cns.count()
print(df)
```

```
  gender GradeID  cns
0      F    G-04    1
1      F    G-06    1
2      F    G-07    3
3      F    G-08    0
4      M    G-04    2
5      M    G-07    6
6      M    G-08    2
```

Write a Python program to display some basic statistical details like percentile, mean, standard deviation etc. of the species of 'Iris-setosa', 'Iris-versicolor' and 'Iris-versicolor' of iris.csv dataset

[40]:

```python
import pandas as pd
import numpy as np
url="Iris.csv"
df = pd.read_csv(url)
df.head(10)
```

[40]:

```
   Id  SepalLengthCm  SepalWidthCm  PetalLengthCm  PetalWidthCm      Species
0   1            5.1           3.5            1.4           0.2  Iris-setosa
1   2            4.9           3.0            1.4           0.2  Iris-setosa
2   3            4.7           3.2            1.3           0.2  Iris-setosa
3   4            4.6           3.1            1.5           0.2  Iris-setosa
4   5            5.0           3.6            1.4           0.2  Iris-setosa
5   6            5.4           3.9            1.7           0.4  Iris-setosa
6   7            4.6           3.4            1.4           0.3  Iris-setosa
7   8            5.0           3.4            1.5           0.2  Iris-setosa
8   9            4.4           2.9            1.4           0.2  Iris-setosa
9  10            4.9           3.1            1.5           0.1  Iris-setosa
```

[41]:

```python
df.describe()
```

[41]:

```
               Id  SepalLengthCm  SepalWidthCm  PetalLengthCm  PetalWidthCm
count  150.000000     150.000000    150.000000     150.000000    150.000000
mean    75.500000       5.843333      3.054000       3.758667      1.198667
std     43.445368       0.828066      0.433594       1.764420      0.763161
min      1.000000       4.300000      2.000000       1.000000      0.100000
25%     38.250000       5.100000      2.800000       1.600000      0.300000
50%     75.500000       5.800000      3.000000       4.350000      1.300000
75%    112.750000       6.400000      3.300000       5.100000      1.800000
max    150.000000       7.900000      4.400000       6.900000      2.500000
```