



 main ▾

...

CLG_CHUTİYAP / DSBDA / 31112_ASSIGNMENT_03.ipynb

 mean-understanding45 *

History

 1 contributor

2199 lines (2199 sloc) | 68.5 KB

...

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. Provide the codes with outputs and explain everything that you do in this step.

```
In [1]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
from sklearn.preprocessing import LabelEncoder
```

```
In [2]: data=pd.read_csv("nba.csv")
```

```
In [3]: data
```

```
Out[3]:
```

	Name	Team	Number	Position	Age	Height	Weight	College	Salary
0	Avery Bradley	Boston Celtics	0.0	PG	25.0	6-2	180.0	Texas	7730337.0
1	Jae Crowder	Boston Celtics	99.0	SF	25.0	6-6	235.0	Marquette	6796117.0
2	John Holland	Boston Celtics	30.0	SG	27.0	6-5	205.0	Boston University	NaN
3	R.J. Hunter	Boston Celtics	28.0	SG	22.0	6-5	185.0	Georgia State	1148640.0
4	Jonas Jerebko	Boston Celtics	8.0	PF	29.0	6-10	231.0	NaN	5000000.0
...
453	Shelvin Mack	Utah Jazz	8.0	PG	26.0	6-3	203.0	Butler	2433333.0
454	Raul Neto	Utah Jazz	25.0	PG	24.0	6-1	179.0	NaN	900000.0
455	Tibor Pleiss	Utah Jazz	21.0	C	26.0	7-3	256.0	NaN	2900000.0
456	Jeff Withey	Utah Jazz	24.0	C	26.0	7-0	231.0	Kansas	947276.0
457	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

458 rows × 9 columns

```
In [4]: data.shape
```

```
Out[4]: (458, 9)
```

```
In [5]: data.head()
```

```
Out[5]:
```

	Name	Team	Number	Position	Age	Height	Weight	College	Salary
0	Avery Bradley	Boston Celtics	0.0	PG	25.0	6-2	180.0	Texas	7730337.0
1	Jae Crowder	Boston Celtics	99.0	SF	25.0	6-6	235.0	Marquette	6796117.0
2	John Holland	Boston Celtics	30.0	SG	27.0	6-5	205.0	Boston University	NaN
3	R.J. Hunter	Boston Celtics	28.0	SG	22.0	6-5	185.0	Georgia State	1148640.0
4	Jonas Jerebko	Boston Celtics	8.0	PF	29.0	6-10	231.0	NaN	5000000.0

```
In [6]: data.dtypes
```

```
Out[6]: Name          object
Team          object
Number        float64
Position      object
Age           float64
Height        object
Weight        float64
College       object
Salary        float64
dtype: object
```

```
In [7]: data.isnull().sum()
```

```
Out[7]: Name          1
Team          1
Number        1
Position      1
Age           1
Height        1
Weight        1
College       85
Salary       12
dtype: int64
```

```
In [8]: mode=data['College'].mode()[0]
data['College'].fillna(mode,inplace=True)
```

```
In [9]: data['Salary'].fillna(data['Salary'].mean(),inplace=True)
```

```
In [10]: data.isnull().sum()
```

```
Out[10]: Name          1
Team          1
Number        1
Position      1
Age           1
Height        1
Weight        1
College       0
Salary       0
dtype: int64
```

```
In [11]: data.shape
```

```
data.shape
```

```
Out[11]: (458, 9)
```

```
In [12]: data.dropna(inplace=True)
```

```
In [13]: data['Height'].value_counts()
```

```
Out[13]: 6-9      59
6-10     47
6-7      45
6-8      43
6-6      42
6-11     40
6-3      33
6-5      32
6-4      29
7-0      27
6-1      16
6-2      16
6-0      10
7-1       7
7-3       4
5-11      3
7-2       3
5-9       1
Name: Height, dtype: int64
```

```
In [14]: heightgroups=data.groupby(data['Height'])
```

```
In [38]: heightgroups.get_group('6-11')
```

```
Out[38]:
```

	Name	Team	Number	Position	Age	Height	Weight	College	Salary
24	Chris McCullough	Brooklyn Nets	1.0	PF	21.0	6-11	200.0	Syracuse	1140240.0
55	Nerlens Noel	Philadelphia 76ers	4.0	PF	22.0	6-11	228.0	Kentucky	3457800.0
56	Jahlil Okafor	Philadelphia 76ers	8.0	C	20.0	6-11	275.0	Duke	4582680.0
60	Christian Wood	Philadelphia 76ers	35.0	PF	20.0	6-11	220.0	UNLV	525093.0
73	Jason Thompson	Toronto Raptors	1.0	PF	29.0	6-11	250.0	Rider	245177.0
81	Festus Ezeli	Golden State Warriors	31.0	C	26.0	6-11	265.0	Vanderbilt	2008748.0
90	Anderson Varejao	Golden State Warriors	18.0	PF	33.0	6-11	273.0	Kentucky	289755.0
91	Cole Aldrich	Los Angeles Clippers	45.0	C	27.0	6-11	250.0	Kansas	1100602.0
98	DeAndre Jordan	Los Angeles Clippers	6.0	C	27.0	6-11	265.0	Texas A&M	19689000.0
113	Ryan Kelly	Los Angeles Lakers	4.0	PF	25.0	6-11	230.0	Duke	1724250.0
143	DeMarcus Cousins	Sacramento Kings	15.0	C	25.0	6-11	270.0	Kentucky	15851950.0
162	Joakim Noah	Chicago Bulls	13.0	C	31.0	6-11	232.0	Florida	13400000.0
163	Bobby Portis	Chicago Bulls	5.0	PF	21.0	6-11	230.0	Arkansas	1391160.0

167	Channing Frye	Cleveland Cavaliers	9.0	PF	33.0	6-11	255.0	Arizona	8193029.0
173	Sasha Kaun	Cleveland Cavaliers	14.0	C	31.0	6-11	260.0	Kansas	1276000.0
188	Andre Drummond	Detroit Pistons	0.0	C	22.0	6-11	279.0	Connecticut	3272091.0
204	Ian Mahinmi	Indiana Pacers	28.0	C	29.0	6-11	250.0	Kentucky	4000000.0
208	Myles Turner	Indiana Pacers	33.0	PF	20.0	6-11	243.0	Texas	2357760.0
209	Shayne Whittington	Indiana Pacers	42.0	PF	25.0	6-11	250.0	Western Michigan	845059.0
211	Giannis Antetokounmpo	Milwaukee Bucks	34.0	SF	21.0	6-11	222.0	Kentucky	1953960.0
216	John Henson	Milwaukee Bucks	31.0	PF	25.0	6-11	229.0	North Carolina	2943221.0
220	Greg Monroe	Milwaukee Bucks	15.0	C	26.0	6-11	265.0	Georgetown	16407500.0
224	Miles Plumlee	Milwaukee Bucks	18.0	C	27.0	6-11	249.0	Duke	2109294.0
237	Zaza Pachulia	Dallas Mavericks	27.0	C	32.0	6-11	275.0	Kentucky	5200000.0
239	Dwight Powell	Dallas Mavericks	7.0	PF	24.0	6-11	240.0	Stanford	845059.0
240	Charlie Villanueva	Dallas Mavericks	3.0	PF	31.0	6-11	232.0	Connecticut	947276.0
251	Dwight Howard	Houston Rockets	12.0	C	30.0	6-11	265.0	Kentucky	22359364.0
294	LaMarcus Aldridge	San Antonio Spurs	12.0	PF	30.0	6-11	240.0	Texas	19689000.0
298	Tim Duncan	San Antonio Spurs	21.0	C	40.0	6-11	250.0	Wake Forest	5250000.0
316	Mike Muscala	Atlanta Hawks	31.0	PF	24.0	6-11	240.0	Bucknell	947276.0
321	Tiago Splitter	Atlanta Hawks	11.0	C	31.0	6-11	245.0	Kentucky	9756250.0
339	Chris Bosh	Miami Heat	1.0	PF	32.0	6-11	235.0	Georgia Tech	22192730.0
373	Marcin Gortat	Washington Wizards	13.0	C	32.0	6-11	240.0	Kentucky	11217391.0
375	Nene Hilario	Washington Wizards	42.0	C	33.0	6-11	250.0	Kentucky	13000000.0
391	Joffrey Lauvergne	Denver Nuggets	77.0	C	24.0	6-11	220.0	Kentucky	1709719.0
399	Gorgui Dieng	Minnesota Timberwolves	5.0	C	26.0	6-11	241.0	Louisville	1474440.0
400	Kevin Garnett	Minnesota Timberwolves	21.0	PF	40.0	6-11	240.0	Kentucky	8500000.0
405	Nikola Pekovic	Minnesota Timberwolves	14.0	C	30.0	6-11	307.0	Kentucky	12100000.0
418	Enes Kanter	Oklahoma City Thunder	11.0	C	24.0	6-11	245.0	Kentucky	16407500.0
439	Mason Plumlee	Portland Trail Blazers	24.0	C	26.0	6-11	235.0	Duke	1415520.0

```
In [16]: heightgroups['Salary'].describe()
```

```
Out[16]:
```

	count	mean	std	min	25%	50%	75%	
Height								
5-11	3.0	5.891553e+05	7.926627e+05	55722.0	133733.0	211744.0	8.558720e+05	15000
5-9	1.0	6.912869e+06	NaN	6912869.0	6912869.0	6912869.0	6.912869e+06	69128
6-0	10.0	5.784075e+06	6.337144e+06	947276.0	2437500.0	3934473.5	4.846419e+06	214680
6-1	16.0	5.217919e+06	4.286013e+06	700902.0	1646160.0	3402626.5	8.633373e+06	135000
6-10	47.0	5.185375e+06	5.063120e+06	222888.0	1054584.5	3815000.0	7.025766e+06	196890
6-11	40.0	6.544397e+06	6.906416e+06	245177.0	1362370.0	3107656.0	1.143804e+07	223590
6-2	16.0	3.523777e+06	3.631376e+06	525093.0	947276.0	1553220.0	4.882013e+06	134370
6-3	33.0	5.821784e+06	5.668225e+06	189455.0	1662360.0	4053446.0	8.000000e+06	200930
6-4	29.0	4.646163e+06	5.275308e+06	134215.0	1015421.0	2525160.0	5.192520e+06	200000
6-5	32.0	4.391786e+06	4.114296e+06	55722.0	1160040.0	3129420.0	6.015152e+06	164070
6-6	42.0	3.586813e+06	4.518975e+06	167406.0	955794.0	1903380.0	4.317674e+06	250000
6-7	45.0	3.504402e+06	4.337857e+06	30888.0	947276.0	1535880.0	4.000000e+06	164070
6-8	43.0	5.950412e+06	6.133934e+06	83397.0	1259700.0	3425510.0	9.321234e+06	229700
6-9	59.0	4.157787e+06	4.517154e+06	111444.0	1053814.0	2500000.0	5.250000e+06	201580
7-0	27.0	5.287712e+06	4.675298e+06	947276.0	2003580.0	4204200.0	7.574380e+06	196890
7-1	7.0	7.400988e+06	6.587462e+06	1175880.0	3441500.0	4950000.0	9.555017e+06	196880
7-2	3.0	6.835639e+06	7.825718e+06	525093.0	2457350.0	4389607.0	9.990912e+06	155920
7-3	4.0	2.307930e+06	1.484918e+06	1000000.0	1150000.0	2050000.0	3.207930e+06	41310

```
In [17]: data.Age.value_counts()
```

```
Out[17]:
```

24.0	47
25.0	45
27.0	41
23.0	41
26.0	36
28.0	31
30.0	31
29.0	28
22.0	26
31.0	22
20.0	19
21.0	19
33.0	14
32.0	13
34.0	10
36.0	10
35.0	9
37.0	4
38.0	4
40.0	3
39.0	2
19.0	2

Name: Age, dtype: int64

```
In [18]: bins= [19,25,31,36,40]
labels = ['19-24','25-30','31-35','36-40']
data['AgeGroup'] = pd.cut(data['Age'], bins=bins,labels=labels, right=False)
```

```
In [35]: data['AgeGroup'].value_counts()
```

```
Out[35]: 25-30    212
19-24    154
31-35     68
36-40     20
Name: AgeGroup, dtype: int64
```

```
In [20]: data.groupby('AgeGroup')['Salary'].mean()
```

```
Out[20]: AgeGroup
19-24    2.761705e+06
25-30    5.870999e+06
31-35    6.635271e+06
36-40    3.897656e+06
Name: Salary, dtype: float64
```

```
In [21]: data.groupby('AgeGroup')['Salary'].median()
```

```
Out[21]: AgeGroup
19-24    1.721380e+06
25-30    4.025000e+06
31-35    4.671342e+06
36-40    2.834470e+06
Name: Salary, dtype: float64
```

```
In [22]: data.groupby('AgeGroup')['Salary'].describe()
```

```
Out[22]:
```

	count	mean	std	min	25%	50%	75%
AgeGroup							
19-24	154.0	2.761705e+06	3.164929e+06	30888.0	1000000.00	1.721380e+06	3.150510e+06
25-30	212.0	5.870999e+06	5.471951e+06	55722.0	1100602.00	4.025000e+06	8.991574e+06
31-35	68.0	6.635271e+06	6.238296e+06	200600.0	2096417.75	4.671342e+06	9.667979e+06
36-40	20.0	3.897656e+06	5.373672e+06	222888.0	947276.00	2.834470e+06	4.276685e+06

◀  ▶

```
In [23]: List_Of_Categories_In_AgeGroup=list(data['AgeGroup'].value_counts().index)
```

```
In [24]: List_Of_Categories_In_AgeGroup
```

```
Out[24]: ['25-30', '19-24', '31-35', '36-40']
```

```
In [25]: list_of_salaries = list(data.groupby('AgeGroup')['Salary'])
```

```
In [26]: list_of_salaries
```

```
Out[26]: [('19-24',
3      1148640.0
6      1170960.0
8      1824360.0
9      3431040.0
10     2569260.0
...
446    12000000.0
447     1175880.0
449     1348440.0
452     2239800.0
454      900000.0
```

```

434      9.000000e+06
      Name: Salary, Length: 154, dtype: float64),
('25-30',
0      7.730337e+06
1      6.796117e+06
2      4.842684e+06
4      5.000000e+06
5      1.200000e+07
...
450     2.050000e+06
451     9.813480e+05
453     2.433333e+06
455     2.900000e+06
456     9.472760e+05
      Name: Salary, Length: 212, dtype: float64),
('31-35',
19      6300000.0
31      1635476.0
33     22875000.0
34      7402812.0
43      947276.0
...
375     13000000.0
394     4345000.0
413     3750000.0
415     3135000.0
434     5016000.0
      Name: Salary, Length: 68, dtype: float64),
('36-40',
46      4.842684e+06
72      2.900000e+06
93      5.675000e+06
101     3.376000e+06
102     9.477260e+05
109     2.500000e+07
119     9.472760e+05
139     1.449187e+06
183     2.170465e+06
236     8.333334e+06
256     9.472760e+05
259     5.000000e+06
260     3.542500e+06
261     4.088019e+06
296     9.472760e+05
299     2.814000e+06
343     2.854940e+06
392     9.472760e+05
406     9.472760e+05
420     2.228880e+05
      Name: Salary, dtype: float64)]

```

```
In [27]: df=pd.read_csv("Iris.csv")
```

```
In [28]: df.head()
```

```
Out[28]:
```

	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

```
In [29]: df.Species.value counts()
```



```
df['Species'].value_counts()
```

```
Out[29]: Iris-setosa      50  
Iris-versicolor    50  
Iris-virginica     50  
Name: Species, dtype: int64
```

```
In [30]: df.shape
```

```
Out[30]: (150, 6)
```

```
In [31]: df.isnull().sum()
```

```
Out[31]: Id      0  
SepalLengthCm  0  
SepalWidthCm   0  
PetalLengthCm  0  
PetalWidthCm   0  
Species        0  
dtype: int64
```

```
In [32]: df.dtypes
```