Kmit

SONET

Data Science

(Level-I)

Ajeet K. Jain, M. Narsimlu

(ML TEAM)- SONET, KMIT, Hyderabad

Session - 22

This session deals with

Exploration Data Analysis

Frequency Tables

Correlation

Data Visualization

Data Preprocessing

Exercise-2

1.Read Toyota cars data and add a new feature to Data frame as Price_Class
based on price value conditions:
a. if price<=8450 then add "Low"
b. if price>11950 then add "High"
c. otherwise add "Medium“
Use While Loop to add each row value to column

Solution


```
import pandas as pd
data_cars=pd.read_csv("ToyotaCorolla.csv")
data_cars.insert(10,"Price_Class","")
i=0
while(i<len(data_cars["Price"])):
    if(data_cars["Price"][i]<=8450):
        data_cars["Price_Class"][i]="Low"
    elif(data_cars["Price"][i]>11950):
        data_cars["Price_Class"][i]="High"
    else:
        data_cars["Price_Class"][i]="Medium"
    i=i+1
print(data_cars["Price_Class"].value_counts())
print(data_cars.columns)
```

Output

```
Medium     751
Low        369
High       316
Name: Price_Class, dtype: int64
Index(['Id', 'Model', 'Price', 'Age_08_04', 'Mfg_Month',
'Mfg_Year', 'KM',
       'Fuel_Type', 'HP', 'Met_Color', 'Price_Class',
'Automatic', 'cc',
       'Doors', 'Cylinders', 'Gears', 'Quarterly_Tax',
'Weight',
       'Mfr_Guarantee', 'BOVAG_Guarantee',
'Guarantee_Period', 'ABS',
       'Airbag_1', 'Airbag_2', 'Airco', 'Automatic_airco',
'Boardcomputer',
       'CD_Player', 'Central_Lock', 'Powered_Windows',
```

Exploration of Data Analysis

Exploratory Data Analysis (EDA) is an approach to analyzing datasets to summarize their main characteristics.

It is used to understand data, get some context regarding it, understand the variables and the relationships between them

It is also useful in formulate hypotheses that could be useful when building predictive models.

The objective is to group records in your data set that have similar categorical attributes and then perform some calculation (count, sum, mean, etc.)

Exploration of Data Analysis

Exploratory Data Analysis (EDA) is actually getting in there, exploring the data, and discovering insights.

One of the fundamental ways to extract insights from a data set is to reduce the size of the data so that you can look at just a piece of it at a time.

There are two ways to do this: *filtering* and *aggregating*.

Essentially removing either rows or columns (or both rows and columns) in order to focus on a subset of the data that interests.

Exploration of Data Analysis

Analyzing the relation between feature variables

Frequency Table

Two-Way tables

Two way table – joint Probability

Two way table – Marginal Probability

Two way table – Conditional Probability

Correlation

One Way Table

Create frequency tables (also known as crosstabs) in pandas using the pd.crosstab() function.

The function takes one or more array-like objects as indexes or columns and then constructs a new DataFrame of variable counts based on the supplied arrays.

Exercise-01

create a frequency table from farms dataset on bimas feature which displays index(it also displays each category count) and columns

```
import pandas as pd
data_farms=pd.read_csv("farms.csv")
bimas_tab = pd.crosstab(index=data_farms["bimas"],
                        columns="count")
print(bimas_tab)
```

Output

```
col_0   count
bimas
mixed     162
no        779
yes        85
```

Exercise -2

Let's make a couple more crosstabs to explore other variables: such as status feature and varieties features

```
import pandas as pd
data_farms=pd.read_csv("farms.csv")
status_tab = pd.crosstab(index=data_farms["status"],
                         columns="count")
varieties_tab = pd.crosstab(index=data_farms["varieties"],
                         columns="count")
print(varieties_tab)
print(status_tab)
```

Output

```
col_0        count
varieties
high           294
mixed           50
trad           682
col_0    count
status
mixed      211
owner      736
share       79
```

One of the most useful aspects of frequency tables is that they allow you to extract the proportion of the data that belongs to each category.

Two Way Table

Two-way frequency tables, also called contingency tables, are tables of counts with two dimensions where each dimension is a different variable.

Two-way tables can give you insight into the relationship between two variables.

To create a two way table, pass two variables to the pd.crosstab() function instead of one

Exercise -3

DATA SCIENCE

create a frequency table between **status VS varieties** features, which shows the relation between two variables.

```
import pandas as pd
data_farms=pd.read_csv("farms.csv")
status_varie=pd.crosstab(index=data_farms["status"],
                         columns=data_farms["varieties"])
print(status_varie)
print("status wise varities of farming")
status_varie.index=["mixed","owner","share"]
print(status_varie)
```

Output

```
varieties  high  mixed  trad
status
mixed        33      7   171
owner       227     41   468
share        34      2    43
status wise varities of farming
varieties  high  mixed  trad
mixed        33      7   171
owner       227     41   468
share        34      2    43
```

Exercise -4

create a frequency table between status VS bimas features,
Perform the following tasks
1.find the indexes in the data frame
2.find the columns in the data frame
3.find the marginal counts (totals for each row and column) by including the argument margins=True

Solution

```
import pandas as pd
data_farms=pd.read_csv("farms.csv")
status_bimas=pd.crosstab(index=data_farms["status"],
                         columns=data_farms["bimas"],margins=True)
print(status_bimas.index)
print(status_bimas.columns)
status_bimas.columns=['mixed', 'no', 'yes',"rowtotal"]
status_bimas.index=['mixed', 'owner', 'share',"coltotal"]
print(status_bimas)
```

Output

```
Index(['mixed', 'owner', 'share', 'All'], dtype='object',
name='status')
Index(['mixed', 'no', 'yes', 'All'], dtype='object', name='bimas')
          mixed   no  yes  rowtotal
mixed        36  146   29       211
owner       119  564   53       736
share         7   69    3        79
coltotal    162  779   85      1026
```

Higher Dimensional Tables

The crosstab() function lets you create tables out of more than two categories.

Higher dimensional tables can be a little confusing to look at, but they can also yield

finer-grained insight into interactions between multiple variables.

Exercise -5

Let's create a 3-way table inspecting status, bimas and region on farming dataset

```
import pandas as pd
data_farms=pd.read_csv("farms.csv")
status_varie_bim=pd.crosstab(index=data_farms["status"],
                             columns=[data_farms["varieties"],
                                      data_farms["bimas"]],
                             margins=True)
print(status_varie_bim)
```

Probability

Probabilities represent the chances of an event 'e' occurring.

In the classic interpretation, a probability is measured by the number of times event 'e' occurs divided by the total number of trials;

In other words, the frequency of the event occurring.

There are three types of probabilities:

1. Joint Probabilities
2. Marginal Probabilities
3. Conditional Probabilities

Two way table – Joint Probability

joint probability which is the probability of two different events occurring at the same time.

Exercise -3

Find the probability of events which occurs both events at the same time i.e. Automatic and Fuel_Type

```
import pandas as pd
data_cars=pd.read_csv("ToyotaCorolla.csv")

Auto_Fuel=pd.crosstab(index=data_cars["Automatic"],
                      columns=data_cars["Fuel_Type"],
                      normalize=True,
                      dropna=True)
print(Auto_Fuel)
```

Output

Fuel_Type	CNG	Diesel	Petrol
Automatic			
0	0.011142	0.107939	0.825209
1	0.000696	0.000000	0.055014

Two way table – Marginal Probability

The interesting thing about a marginal probability is that the term sounds complicated, but it's actually the probability that we are most familiar with.

Basically anytime you are in interested in a single event irrespective of any other event (i.e. "marginalizing the other event"), then it is a marginal probability.

Exercise -4

Find the probability of event which occurs Automatic and Fuel_Type as single event

```
import pandas as pd
data_cars=pd.read_csv("ToyotaCorolla.csv")

Auto_Fuel=pd.crosstab(index=data_cars["Automatic"],
                      columns=data_cars["Fuel_Type"],
                      margins=True,
                      normalize=True,
                      dropna=True)
print(Auto_Fuel)
```

Output

Fuel_Type	CNG	Diesel	Petrol	All
Automatic				
0	0.011142	0.107939	0.825209	0.94429
1	0.000696	0.000000	0.055014	0.05571
All	0.011838	0.107939	0.880223	1.00000

Two way table – Conditional Probability

A conditional probability is the probability of an event X occurring when a secondary event Y is true.

Mathematically, it is represented as P(X | Y). This is read as "probability of X given/conditioned on Y".

e. A conditional probability can be calculated as follows:

$$P(X \mid Y) = \frac{P(X, Y)}{P(Y)}$$

Exercise-6

Given type of gear box, probability of different fuel type

```
import pandas as pd
data_cars=pd.read_csv("ToyotaCorolla.csv")

Auto_Fuel=pd.crosstab(index=data_cars["Automatic"],
                      columns=data_cars["Fuel_Type"],
                      margins=True,
                      normalize='index',
                      dropna=True)
print(Auto_Fuel)
```

Output

Fuel_Type	CNG	Diesel	Petrol
Automatic			
0	0.011799	0.114307	0.873894
1	0.012500	0.000000	0.987500
All	0.011838	0.107939	0.880223

Correlation

- **Correlation** – show whether and how strongly pairs of variables are related.
  - For Example**:** engine size, price are related; *A decent prediction of price can be made using engine size.*
  - **Correlation can tell you just how much of the variation in products' engine size is related to price.**

Correlation

- In general, r > 0 – positive relationship
- r<0 – negative relationship
- r = 0 – no relationship (meaning the variables are independent and not related)
- when r = +1.0 – describes a perfect +ve correlation
- when r = -1.0 – describes a perfect -ve correlation
- closer the coefficients are to +1.0 and -1.0, greater is the strength of the relationship between the variables.

Exercise-6

Find the Correlation between numerical variables

```
import pandas as pd
data_cars=pd.read_csv("ToyotaCorolla.csv")


numerical_data=data_cars.select_dtypes(exclude=["object"])
corr_matrix=numerical_data.corr()
print(corr_matrix.head())
```

Output

	Id	Price	...	Radio_cassette	Tow_Bar
Id	1.000000	-0.738250	...	-0.011611	0.159171
Price	-0.738250	1.000000	...	-0.043179	-0.172369
Age_08_04	0.906132	-0.876590	...	0.012857	0.188720
Mfg_Month	0.043742	-0.018138	...	0.032576	-0.042170
Mfg_Year	-0.919523	0.885159	...	-0.018844	-0.182206

[5 rows x 35 columns]

Conclusion

You are aware of

Pandas

EDA

We will proceed with

Data Visualization

Kmit

SONET

DATA SCIENCE

THANK YOU