

▼ Contingency Table

Estimations like mean, median, standard deviation, and variance are very much useful in case of the univariate data analysis.

But in the case of bivariate analysis(comparing two variables) correlation comes into play.

Contingency Table

It is one of the techniques for exploring two or even more variables.

It is basically a tally of counts between two or more categorical variables.

Contingency Tables are giving clear correlation values between two and more variables.

Thus making it much more useful to understand the data for further information extraction

```
1 import numpy as np
2 import pandas as pd
3 import matplotlib as plt
4
```

```
1 data = pd.read_csv("loan_status.csv")
2
3 print (data.head(10))
4
5
```

	grade	sub_grade	loan_status	purpose
0	B	B2	Fully Paid	credit_card
1	C	C4	Charged Off	car
2	C	C5	Fully Paid	small_business
3	C	C1	Fully Paid	other
4	B	B5	Fully Paid	other
5	A	A4	Fully Paid	wedding
6	C	C5	Fully Paid	debt_consolidation
7	E	E1	Fully Paid	car

```

8      F      F2 Charged Off      small_business
9      R      R5 Charged Off      other

```

```
1 data.describe()
```

```
2
```

	grade	sub_grade	loan_status	purpose
count	50	50	50	50
unique	6	19	2	10
top	B	B3	Fully Paid	debt_consolidation
freq	21	6	39	22

```
1 # data types of feature/attributes
```

```
2 # in the data
```

```
3 data.dtypes
```

```
4
```

```

grade      object
sub_grade   object
loan_status object
purpose     object
dtype: object

```

pandas.crosstab() function in Python

This method is used to compute a simple cross-tabulation of two (or more) factors.

By default, computes a frequency table of the factors unless an array of values and an aggregation function are passed.

Syntax: `pandas.crosstab(index, columns, values=None, rownames=None, colnames=None, aggfunc=None, margins=False, margins_name='All', dropna=True, normalize=False)`

Arguments :

index : array-like, Series, or list of arrays/Series, Values to group by in the rows.

columns : array-like, Series, or list of arrays/Series, Values to group by in the columns.

values : array-like, optional, array of values to aggregate according to the factors.

Requires **aggfunc** be specified. **rownames** : sequence, default None, If passed, must match number of row arrays passed.

colnames : sequence, default None, If passed, must match number of column arrays passed.

aggfunc : function, optional, If specified, requires **values** be specified as well. **margins** : bool, default False, Add row/column margins (subtotals).

margins_name : str, default 'All', Name of the row/column that will contain the totals when margins is True.

dropna : bool, default True, Do not include columns whose entries are all NaN.

▼ Contingency Table showing correlation between Grades and loan status.

```
1 data_crosstab = pd.crosstab(data['grade'],
2                             data['loan_status'],
3                             margins = False)
4 print(data_crosstab)
5
```

	loan_status	Charged Off	Fully Paid
grade			
A		1	11
B		5	16
C		3	8
D		1	3
E		0	1
F		1	0

▼ Contingency Table showing correlation between Purpose and loan status.

```

1 data_crosstab = pd.crosstab(data['purpose'],
2                             data['loan_status'],
3                             margins = False)
4 print(data_crosstab)
5

```

loan_status	Charged Off	Fully Paid
purpose		
car	1	1
credit_card	0	8
debt_consolidation	4	18
home_improvement	0	1
major_purchase	1	1
medical	0	1
moving	0	1
other	4	5
small_business	1	2
wedding	0	1

Contingency Table showing correlation between Grades+Purpose and loan status.

```

1 data_crosstab = pd.crosstab([data.grade, data.purpose],
2                             data.loan_status, margins = False)
3 print(data_crosstab)
4

```

loan_status	Charged Off	Fully Paid
grade purpose		
A credit_card	0	1
debt_consolidation	1	7
major_purchase	0	1
other	0	1
wedding	0	1
B credit_card	0	6
debt_consolidation	1	5

	major_purchase	1	0
	medical	0	1
	moving	0	1
	other	3	2
	small_business	0	1
C	car	1	0
	credit_card	0	1
	debt_consolidation	2	4
	home_improvement	0	1
	other	0	1
	small_business	0	1
D	debt_consolidation	0	2
	other	1	1
E	car	0	1
F	small_business	1	0

1

