

## Percentage Table and Contingency Table in Bivariate Analysis

### 1. Percentage Table

A **percentage table** represents data in terms of percentages, making it easier to analyze proportions and trends across categories. It is typically derived from a contingency table.

- **Use Case:** Analyzing the relative frequency distribution of two categorical variables.

### 2. Contingency Table

A **contingency table** (or cross-tabulation) summarizes the relationship between two categorical variables. It displays frequencies (counts) for combinations of the variables.

- **Use Case:** Exploring relationships and dependencies between two variables (e.g., gender and preference).

### Syntax for Creating Tables

1. **Contingency Table:**
  - Use `pandas.crosstab()` to compute frequencies for two variables.
2. **Percentage Table:**
  - Normalize a contingency table to compute row-wise or column-wise percentages.

### Example Code

#### *Creating a Contingency Table*

```
import pandas as pd
# Sample dataset
data = {
    'Gender': ['Male', 'Female', 'Female', 'Male', 'Male', 'Female', 'Male', 'Female'],
    'Preference': ['Tea', 'Coffee', 'Tea', 'Coffee', 'Tea', 'Tea', 'Coffee', 'Coffee']
}
```

```
# Create a DataFrame
df = pd.DataFrame(data)

# Generate Contingency Table
contingency_table = pd.crosstab(df['Gender'], df['Preference'])
print("Contingency Table:\n", contingency_table)
```

#### *Creating a Percentage Table*

```
# Generate Percentage Table (Row-wise normalization)
percentage_table = pd.crosstab(df['Gender'], df['Preference'], normalize='index') * 100
print("\nPercentage Table (Row-wise):\n", percentage_table)
```

**1. Dataset:**

- Gender: Categorical variable with two categories (Male, Female).
- Preference: Categorical variable with two categories (Tea, Coffee).

**2. Contingency Table:**

- `pd.crosstab()` generates a table where:
  - Rows represent Gender.
  - Columns represent Preference.
  - Values represent the count of occurrences for each combination.

**3. Percentage Table:**

- `normalize='index'`: Converts counts to percentages for each row (e.g., percentage of males preferring tea or coffee).

**Output****1. Contingency Table:**

```
Contingency Table:
Preference Coffee Tea
Gender
Female      2  2
Male       2  2
```

**2. Percentage Table (Row-wise):**

```
Percentage Table (Row-wise):
Preference  Coffee  Tea
Gender
Female     50.0   50.0
Male      50.0   50.0
```

**Adding Margins (Totals)**

```
# Contingency Table with Margins
```

```
contingency_with_margins = pd.crosstab(df['Gender'], df['Preference'], margins=True)
print("\nContingency Table with Totals:\n", contingency_with_margins)
```

**Output:**

```
Contingency Table with Totals:
Preference Coffee Tea All
Gender
Female     2  2  4
Male      2  2  4
All       4  4  8
```

## Visualization

### *Bar Plot of Contingency Table*

```
import matplotlib.pyplot as plt
# Plot Contingency Table
contingency_table.plot(kind='bar', stacked=True, figsize=(8, 6), color=['skyblue', 'orange'])
plt.title("Contingency Table: Gender vs. Preference")
plt.xlabel("Gender")
plt.ylabel("Count")
plt.legend(title="Preference")
plt.grid(axis='y')
plt.show()
```

### *Pie Chart of Percentage Table*

```
# Plot Row-wise Percentage as Pie Chart for Females
female_percentage = percentage_table.loc['Female']
female_percentage.plot.pie(
    autopct='%1.1f%%',
    figsize=(6, 6),
    title="Preference Percentage for Female",
    colors=['skyblue', 'orange']
)
plt.ylabel("") # Remove y-label for better visualization
plt.show()
```

#### 1. Bar Plot:

- Shows the distribution of preferences for each gender.
- Stacked bars help compare total counts and individual contributions.

#### 2. Pie Chart:

- Highlights percentage breakdown of preferences for a specific category (e.g., Female).