PROJECT REPORT ON

# EFFECTIVENESS OF DRUG EVALUATION


| | |
|---|---|
| Keshwatkar Shravantika | 1051-22-508-003 |
| Palakurthy Ashritha | 1051-22-508-008 |
| Pyata Sai Prasanna | 1051-22-508-013 |
| Challa Ajay | 1051-22-508-018 |
| Kuruvella Hiranmai | 1051-22-508-023 |


Project submitted for the award of the degree of

**MASTER OF SCIENCE**

BY

OSMANIA UNIVERSITY, HYDERABAD – 500007

## Aurora's Degree & PG College

**Accredited by NAAC with 'B++' grade**

**Department of Statistics, Chikkadpally - 500020**

# EFFECTIVENESS OF DRUG EVALUATION

This project work dissertation is to submitted to

Osmania University, as partial fulfilment for the completion of

M.Sc. III semester of M.Sc. (Applied Statistics)

**By**


| | |
|---|---|
| Keshwatkar Shravantika | 1051-22-508-003 |
| Palakurthy Ashritha | 1051-22-508-008 |
| Pyata Sai Prasanna | 1051-22-508-013 |
| Challa Ajay | 1051-22-508-018 |
| Kuruvella Hiranmai | 1051-22-508-023 |



Under the supervision of

**Mrs.Y. Aruna Rekha**

**Department of Statistics, Aurora's Degree & PG college**

**2022-2024**

**DATE:20/02/2024**

## DECLARATION

I hereby declare the project work presented in the dissertation, entitled **"EFFECTIVENESS OF DRUG EVALUATION"** has been carried out by us under the supervision of **Mrs.Y. Aruna Rekha**, in the Department of Statistics, Aurora's Degree and PG College, Chikkadpally, Hyderabad – 500020. The work done is original and has not been submitted so far, in part or full, for any other degree or diploma to any University or institution.

| | |
|---|---|
| Keshwatkar Shravantika | 1051-22-508-003 |
| Palakurthy Ashritha | 1051-22-508-008 |
| Pyata Sai Prasanna | 1051-22-508-013 |
| Challa Ajay | 1051-22-508-018 |
| Kuruvella Hiranmai | 1051-22-508-023 |

# Aurora's Degree & PG College

**accredited by NAAC with 'B++' grade**

**DATE:20/02/2024**

## CERTIFICATE

This is to certify that the research work presented in this thesis, entitled **"EFFECTIVENESS OF DRUG EVALUATION"** has been carried out by

| | |
|---|---|
| Keshwatkar Shravantika | 1051-22-508-003 |
| Palakurthy Ashritha | 1051-22-508-008 |
| Pyata Sai Prasanna | 1051-22-508-013 |
| Challa Ajay | 1051-22-508-018 |
| Kuruvella Hiranmai | 1051-22-508-023 |

registered M.Sc. (Applied Statistics) students of this college. The work done is original and has not been submitted so far, in part or full, for any other degree or diploma to any University or institution.

**Signature of Head with office seal** **Signature of the Supervisor**

Internal Examiner Signature with date:

External Examiner Signature with date:

## ACKNOWLEDGMENT

I Am thankful to my supervisor **Mrs.Y. Aruna Rekha**, Assistant Professor**, Statistics Department, Aurora's Degree & PG College, Chikkadpally, Hyderabad – 500020,** for her guidance.

I would like to express my sincere thanks to all the Faculty, **Department of Statistics, Aurora's Degree & PG College** for encouragement throughout the period of our coursework.

| | |
|---|---|
| Keshwatkar Shravantika | 1051-22-508-003 |
| Palakurthy Ashritha | 1051-22-508-008 |
| Pyata Sai Prasanna | 1051-22-508-013 |
| Challa Ajay | 1051-22-508-018 |
| Kuruvella Hiranmai | 1051-22-508-023 |

# ABSTRACT

The script starts by loading a dataset from a CSV file, containing information about drugs, their indications, effectiveness, ease of use, satisfaction, and other relevant attributes. Data cleaning and preprocessing steps are performed, including handling duplicates, dropping unnecessary columns, and categorizing numerical values into meaningful categories.

Exploratory Data Analysis (EDA) is conducted to analyze the distribution of drug types, conditions, and other features. Missing values are checked, and descriptive statistics are calculated. Categorical variables like 'Effective_Cat,' 'EaseOfUse_Cat,' and 'Satisfaction_Cat' are created based on predefined criteria.

The script then utilizes visualization techniques using “seaborn” and “plotly.” express to generate bar plots, heatmaps, and scatter plots to visualize relationships and trends in the data. Correlation analysis is performed using a heatmap.

The code proceeds to build a linear regression model using scikit-learn, predicting drug effectiveness based on ease of use and satisfaction levels. Model performance is evaluated using mean squared error, and the predicted effectiveness values are compared with actual values through a bar plot.

In summary, the script combines data preprocessing, exploratory data analysis, and machine learning techniques to analyze and visualize drug-related information, providing insights into the relationships between drug attributes and their effectiveness.

# INDEX

## TITLE:EFFECTIVENESS OF DRUG EVALUATION

# CHAPTER- 1
# INTRODUCTION

## 1.1 Objective

The goal is to analyze a dataset on drugs and understand how factors like ease of use and patient satisfaction relate to drug effectiveness. Through data cleaning, visualization, and a simple predictive model, we aim to find patterns and insights that could be valuable for decision-making in the healthcare domain.

## 1.2 Source of Data

https://www.kaggle.com/datasets/thedevastator/drug-performance-evaluation.

## 1.3 Data Description

The effectiveness of drug evaluation plays a pivotal role in shaping healthcare practices and patient well-being. Assessing how drugs perform in real-world scenarios involves a multidimensional analysis that considers factors such as efficacy, ease of use, and patient satisfaction. This evaluation process is instrumental in guiding healthcare professionals, researchers, and regulatory bodies in making informed decisions regarding the adoption and continued use of pharmaceutical interventions.

Understanding the effectiveness of drug evaluation involves delving into comprehensive datasets that capture diverse aspects of drug performance. By examining the interplay between variables like ease of use and patient satisfaction, we aim to unravel insights that contribute to the optimization of therapeutic outcomes.

In this exploration, data analysis techniques, visualization tools, and predictive modeling are employed to unveil patterns and correlations within the dataset. The goal is to provide a

nuanced understanding of the factors influencing the effectiveness of drug evaluation, fostering a data-driven approach to healthcare decision-making.

Through this analysis, we aim to contribute valuable perspectives to the ongoing discourse on drug efficacy, ultimately striving for advancements that enhance the overall quality of patient care and treatment outcomes.

# CHAPTER- 2
# LITERATURE REVIEW

## Introduction:

In the contemporary healthcare landscape, the evaluation of drug effectiveness, ease of use, and patient satisfaction plays a pivotal role in enhancing patient outcomes and overall healthcare quality. Understanding how different drugs perform for specific medical conditions and how patients perceive their ease of use and satisfaction is crucial for informed decision-making by healthcare providers.

### ➢ Drug Effectiveness

Research on drug effectiveness spans various therapeutic areas, with a focus on assessing the ability of drugs to address specific medical conditions. Clinical trials and observational studies often contribute to the body of evidence supporting or refuting the efficacy of drugs. Factors influencing drug effectiveness include pharmacokinetics, pharmacodynamics, and the interaction with the targeted pathophysiological mechanisms.

### ➢ Ease of Use

The ease of use of medications is a critical aspect of patient adherence and overall treatment success. Studies in this domain explore how the formulation, route of administration, and dosage frequency impact patients' ability to adhere to prescribed regimens. Factors such as pill burden, complexity of administration, and the integration of medication into patients' daily routines are commonly examined to determine the ease of use.

### ➢ Patient Satisfaction

Patient satisfaction is a multifaceted concept that encompasses various dimensions, including treatment efficacy, side effects, and overall experience with the prescribed drug. Understanding

patient perspectives provides insights into treatment adherence, quality of life, and the patient's willingness to continue the prescribed regimen. Patient-reported outcomes and surveys are commonly employed to gauge satisfaction levels.

- **Integration of Data**

The dataset provided combines information on drug effectiveness, ease of use, and patient satisfaction for a diverse set of medical conditions. This integration allows for a comprehensive analysis of the interplay between these variables and provides a holistic view of the patient experience with different medications.

- **Challenges and Opportunities**

Variability in individual responses, potential biases in patient reporting, and the dynamic nature of healthcare environments may influence the interpretation of results. Despite these challenges, the dataset presents an opportunity to identify patterns While the dataset offers valuable insights, it is important to acknowledge potential, trends, and potential areas for improvement in drug development and patient care.

# CHAPTER- 3
# METHODOLOGY

## 3.1 Data Collection

The dataset used in this analysis was sourced from https://www.kaggle.com/datasets/thedevastator/drug-performance-evaluation.. The dataset includes information on drug effectiveness, ease of use, and patient satisfaction across a range of medical conditions. The data was collected using Data Analysis of Drug Effectiveness and Evaluation. The dataset contains 2219,8 entries.

## 3.2 Variables

Condition: The medical condition for which the drug is prescribed.

Drug: The specific drug prescribed for the given condition.

Indication: The indication, whether the drug is labeled as "On Label" or "Off Label."

Type: The type of drug, categorized as "RX" (prescription), "OTC" (over-the-counter), or "RX/OTC."

Reviews: The number of reviews or feedback received for each drug.

Effective: Numerical rating indicating the effectiveness of the drug.

EaseOfUse: Numerical rating indicating the ease of use of the drug.

Satisfaction: Numerical rating indicating patient satisfaction

## 3.3 Data Preprocessing

Before analysis, several steps were taken to preprocess the data:

Handling Duplicates: Duplicate entries were removed to ensure the integrity of the dataset.

Handling Missing Values: Any missing or irrelevant values were addressed appropriately.

Categorization: Ratings for effectiveness, ease of use, and satisfaction were categorized for easier interpretation and analysis.

## 3.4 Exploratory Data Analysis (EDA)

EDA was conducted to understand the distribution of variables, identify outliers, and explore potential patterns. Visualizations such as bar plots, box plots, and scatter plots were employed to gain insights into the relationships between variables.

## 3.5 Categorical Variables Analysis

Categorical variables such as drug type, drug indication, and effectiveness category were analyzed to observe any significant trends or patterns. This involved using counts, percentages, and visualizations.

## 3.6 Correlation Analysis

Correlation analysis was performed to identify potential relationships between numerical variables, particularly focusing on the correlation between drug effectiveness, ease of use, and patient satisfaction.

## 3.7 Predictive Modeling

Linear regression modeling was employed to predict drug satisfaction based on ease of use and other relevant variables. The model's performance was evaluated using mean squared error.

## 3.8 Data Visualization

Data visualization played a crucial role in presenting key findings and insights. Visualizations were created using libraries such as Matplotlib, Seaborn, and Plotly to enhance the interpretability of results.

## 3.9 Linear Regression:

Linear regression is a statistical method used to model the relationship between a dependent variable (target) and one or more independent variables (predictors). It assumes a linear relationship between the independent variables and the dependent variable. Here's an explanation of linear regression modeling.

- **Assumptions:**
    - Linearity: The relationship between the independent and dependent variables is assumed to be linear.
    - Independence: Observations are assumed to be independent of each other.
    - Homoscedasticity: The variance of the errors is constant across all levels of the independent variables.
    - Normality: The residuals (the differences between observed and predicted values) are assumed to be normally distributed.

## ➢ Model Representation:

- In simple linear regression, there is one independent variable and one dependent variable, represented as $y=\beta 0+\beta 1x+\epsilon$, where:
  - $y$ is the dependent variable.
  - $x$ is the independent variable.
  - $\beta 0$ is the intercept (the value of $y$ when $x=0$).
  - $\beta 1$ is the slope (the change in $y$ for a one-unit change in $x$).
  - $\epsilon$ is the error term (the difference between the observed and predicted values of $y$).

## ➢ Model Fitting:

- The goal of linear regression is to find the best-fitting line (or hyperplane in the case of multiple predictors) that minimizes the sum of squared differences between the observed and predicted values of the dependent variable.
- This is typically done using the method of least squares, where the parameters $\beta 0$ and $\beta 1$ are estimated to minimize the residual sum of squares (RSS).

## ➢ Model Evaluation:

- After fitting the model, it's important to assess its performance. Common evaluation metrics include:
  - Mean Squared Error (MSE): Measures the average squared difference between observed and predicted values.

## ➢ Interpretation:

- Once the model is fitted, the coefficients ($\beta 0$ and $\beta 1$) provide insights into the relationship between the independent and dependent variables. For example, in a simple linear regression, the slope coefficient ($\beta 1$) represents the change in the dependent variable for a one-unit change in the independent variable.

In summary, linear regression is a powerful tool for modeling and understanding the relationship between variables. It provides insights into how changes in one variable are associated with changes in another, allowing for predictions and inference. However, it's important to assess model assumptions and evaluate performance to ensure the reliability of the results.

We are implementing linear regression modeling to predict drug effectiveness based on ease of use and satisfaction ratings. It involves data preparation, feature selection, model training, evaluation, interpretation of results, and prediction. The goal is to understand how changes in ease of use and satisfaction levels affect drug effectiveness and make predictions for new drugs.

## ➢ Data Preparation:

- The code first preprocesses the dataset, including dropping irrelevant columns, handling missing values, removing duplicates, and categorizing numerical variables into categories.

## ➢ Feature Selection:

- The independent variables (features) selected for the linear regression model are 'EaseOfUse' and 'Satisfaction'. These features are chosen based on their potential to explain or predict the dependent variable 'Effective', which represents drug effectiveness.

## ➢ Data Splitting:

- The dataset is split into training and testing sets using the **train_test_split** function from sklearn. The training set is used to train the linear regression model, while the testing set is used to evaluate its performance.

## ➢ Model Training:

- The linear regression model is trained on the training data using the **Linear Regression** class from sklearn. The **fit** method is called on the model object, passing the independent variables (features) and the dependent variable (target) as arguments

## ➢ Model Evaluation:

- After training the model, it's evaluated using the testing data. The model predicts the target variable ('Effective') for the testing data using the **predict** method. Then, the mean squared error (MSE) is calculated to assess the performance of the model. Lower MSE values indicate better model performance.

## ➢ Interpretation:

- The model coefficients (intercept and slope) are printed to understand the relationship between the independent variables ('EaseOfUse' and 'Satisfaction')

and the dependent variable ('Effective'). The intercept represents the predicted value of 'Effective' when both 'EaseOfUse' and 'Satisfaction' are zero. The slope coefficients represent the change in 'Effective' for a one-unit change in 'EaseOfUse' and 'Satisfaction'.

## ➢ Prediction:

- Finally, the model predicts the satisfaction level for a hypothetical new drug with ease-of-use level 4 and satisfaction level 5. These values are passed to the model's **predict** method, and the predicted satisfaction level is printed.

# CHAPTER- 4
# DATA ANALYSIS

**DATA CLEANING**

## ❖ Importing Libraries and Reading the Dataset:

- The script begins by importing necessary libraries such as **pandas, numpy, seaborn,** matplotlib**. pyplot, sklearn, and plotly. express**. These libraries are commonly used for data manipulation, visualization, and machine learning tasks.
- The code reads a CSV file named 'Mini project data.csv' into a pandas DataFrame called **df** using the **pd. read_csv ()** function. This DataFrame likely contains information about different drugs, their indications, effectiveness, ease of use, satisfaction, and other relevant features.

```
In [1]: #Importing the libraries
        import pandas as pd
        import numpy as np
        import seaborn as sns
        import matplotlib.pyplot as plt

In [2]: import sklearn

In [3]: df=pd.read_csv(r'C:\Users\Hiranmail1\OneDrive\Desktop\Mini project data.csv')

In [4]: df.head()

Out[4]:
```

| | Condition | Drug | Indication | Type | Reviews | Effective | EaseOfUse | Satisfaction | Unnamed: 8 |
|---|---|---|---|---|---|---|---|---|---|
| 0 | Acute Bacterial Sinusitis | Levofloxacin | On Label | RX | 994 Reviews | 2.52 | 3.01 | 1.84 | \r\n\t\t\t\t\tLevofloxacin is used to treat a ... |
| 1 | Acute Bacterial Sinusitis | Levofloxacin | On Label | RX | 994 Reviews | 2.52 | 3.01 | 1.84 | \r\n\t\t\t\t\tLevofloxacin is used to treat a ... |
| 2 | Acute Bacterial Sinusitis | Moxifloxacin | On Label | RX | 755 Reviews | 2.78 | 3.00 | 2.08 | \r\n\t\t\t\t\t This is a generic drug. The ave... |
| 3 | Acute Bacterial Sinusitis | Azithromycin | On Label | RX | 584 Reviews | 3.21 | 4.01 | 2.57 | \r\n\t\t\t\t\tAzithromycin is an antibiotic (m... |
| 4 | Acute Bacterial Sinusitis | Azithromycin | On Label | RX | 584 Reviews | 3.21 | 4.01 | 2.57 | \r\n\t\t\t\t\tAzithromycin is an antibiotic (m... |

## ❖ Data Cleaning and Preprocessing:

- Several data cleaning and preprocessing steps are performed to prepare the dataset for analysis. This includes dropping unnecessary columns (like 'Unnamed: 8'), handling missing values, dropping duplicate rows, rounding numerical columns to two decimal

places, and categorizing numerical variables into meaningful categories based on thresholds.

```
In [5]: df.drop(columns = ['Unnamed: 8'], axis = 1, inplace = True)

In [6]: df.head()
```

Out[6]:

| | Condition | Drug | Indication | Type | Reviews | Effective | EaseOfUse | Satisfaction |
|---|---|---|---|---|---|---|---|---|
| 0 | Acute Bacterial Sinusitis | Levofloxacin | On Label | RX | 994 Reviews | 2.52 | 3.01 | 1.84 |
| 1 | Acute Bacterial Sinusitis | Levofloxacin | On Label | RX | 994 Reviews | 2.52 | 3.01 | 1.84 |
| 2 | Acute Bacterial Sinusitis | Moxifloxacin | On Label | RX | 755 Reviews | 2.78 | 3.00 | 2.08 |
| 3 | Acute Bacterial Sinusitis | Azithromycin | On Label | RX | 584 Reviews | 3.21 | 4.01 | 2.57 |
| 4 | Acute Bacterial Sinusitis | Azithromycin | On Label | RX | 584 Reviews | 3.21 | 4.01 | 2.57 |

- Dropping the "\r\n" rows since its row id is not properly defined.

```
In [13]: # droping the '\r\n' rows since its name is properly not defined
         df = df.drop(df[df['Type'] == '\r\n' ].index)

In [14]: df['Type'].value_counts()

Out[14]: RX        1165
         OTC        546
         RX/OTC      34
         Name: Type, dtype: int64
```

```
In [21]: df = df.drop(df[df['Indication'] == '\r\n' ].index)

In [22]: df['Indication'].value_counts()

Out[22]: On Label     1331
         Off Label     384
         Name: Indication, dtype: int64
```

- Checking missing values.

```
In [38]: #Checking missing values:
         df.isna().sum()

Out[38]: Condition           0
         Drug                0
         Indication          0
         Type                0
         Reviews             0
         Effective           0
         EaseOfUse           0
         Satisfaction        0
         Effective_Cat       0
         EaseOfUse_Cat       0
         Satisfaction_Cat    0
         dtype: int64
```

- Categorizing numerical into categories.

```
In [25]: def categorize_effectiveness(effective):
             if effective < 2.0 : return 'Ineffective'
             elif effective < 3.0 : return 'Partly Effective'

             elif effective <= 5.0 : return 'Effective'
```

```
In [30]: def categorize_ease_of_use(easeOfUse):
             if easeOfUse < 2.0 : return 'Difficult'
             elif easeOfUse < 3.0 : return 'Normal'
             elif easeOfUse <= 5.0 : return 'Easy'
```

```
In [34]: def categorize_satisfaction(satisfaction):
             if satisfaction < 2.0 : return 'Unsatisfied'
             elif satisfaction < 3.0 : return 'Partly Satisfied'
             elif satisfaction <= 5.0 : return 'Very Satisfied'
```

- Categorized data.

```
In [40]: df.drop_duplicates(keep='first', inplace=True)
         df.reset_index(drop=True)
```

Out[40]:

| | Condition | Drug | Indication | Type | Reviews | Effective | EaseOfUse | Satisfaction | Effective_Cat | EaseOfUse_Cat | Satisfaction_Cat |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | Acute Bacterial Sinusitis | Levofloxacin | On Label | RX | 994 Reviews | 2.52 | 3.01 | 1.84 | Partly Effective | Easy | Unsatisfied |
| 1 | Acute Bacterial Sinusitis | Moxifloxacin | On Label | RX | 755 Reviews | 2.78 | 3.00 | 2.08 | Partly Effective | Easy | Partly Satisfied |
| 2 | Acute Bacterial Sinusitis | Azithromycin | On Label | RX | 584 Reviews | 3.21 | 4.01 | 2.57 | Effective | Easy | Partly Satisfied |
| 3 | Acute Bacterial Sinusitis | Amoxicillin-Pot Clavulanate | On Label | RX | 437 Reviews | 3.26 | 3.23 | 2.42 | Effective | Easy | Partly Satisfied |
| 4 | Acute Bacterial Sinusitis | Levofloxacin | On Label | RX | 361 Reviews | 2.44 | 2.96 | 1.68 | Partly Effective | Normal | Unsatisfied |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 1710 | vulvovaginal candidiasis | Clotrimazole | On Label | OTC | 2 Reviews | 5.00 | 5.00 | 5.00 | Effective | Easy | Very Satisfied |
| 1711 | vulvovaginal candidiasis | Butoconazole Nitrate | On Label | RX | 1 Reviews | 5.00 | 5.00 | 5.00 | Effective | Easy | Very Satisfied |
| 1712 | vulvovaginal candidiasis | Clotrimazole | On Label | OTC | 1 Reviews | 5.00 | 4.00 | 5.00 | Effective | Easy | Very Satisfied |
| 1713 | vulvovaginal candidiasis | Butoconazole Nitrate | On Label | OTC | 1 Reviews | 5.00 | 5.00 | 5.00 | Effective | Easy | Very Satisfied |
| 1714 | vulvovaginal candidiasis | Miconazole-Skin Clnsr17 | On Label | OTC | 1 Reviews | 1.00 | 4.00 | 1.00 | Ineffective | Easy | Unsatisfied |

1715 rows × 11 columns

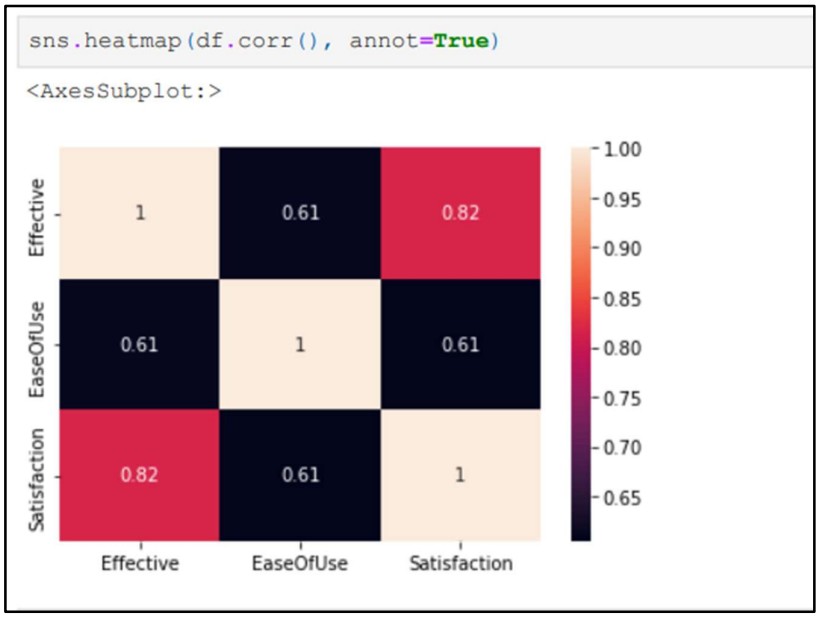
# CHAPTER- 5
# GRAPHICAL REPRESENTATION

## 5.1 Heat map

The heatmap provides a graphical representation of the correlations between different pairs of variables in the DataFrame. The correlation coefficient measures the strength and direction of the linear relationship between two variables. The value ranges from -1 to 1, where:

- 1 indicates a perfect positive correlation (as one variable increases, the other variable also increases),
- -1 indicates a perfect negative correlation (as one variable increases, the other variable decreases), and
- 0 indicates no correlation between the variables.

The intensity of the color in each cell of the heatmap represents the strength of the correlation between the corresponding pair of variables. Positive correlations are typically represented by lighter colors, while negative correlations are represented by darker colors.

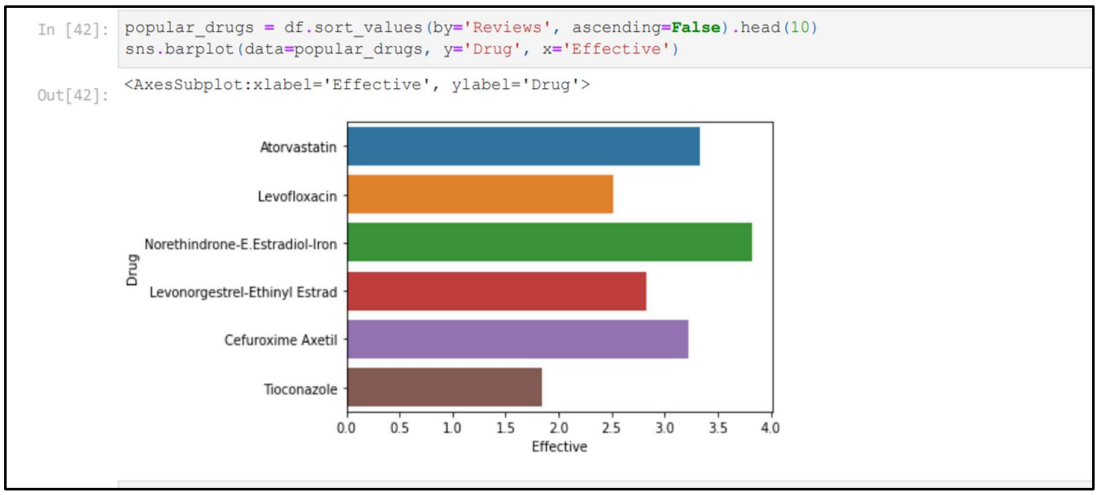


## 5.2 Bar plots

- The bar plot using Seaborn **(sns.barplot)** to visualize the effectiveness **(Effective)** of these popular drugs.

- The y-axis of the bar plot represents the names of the drugs (**Drug**).
- The x-axis represents the effectiveness of the drugs (**Effective**).

This visualization helps to understand the effectiveness of the most reviewed drugs in the dataset.

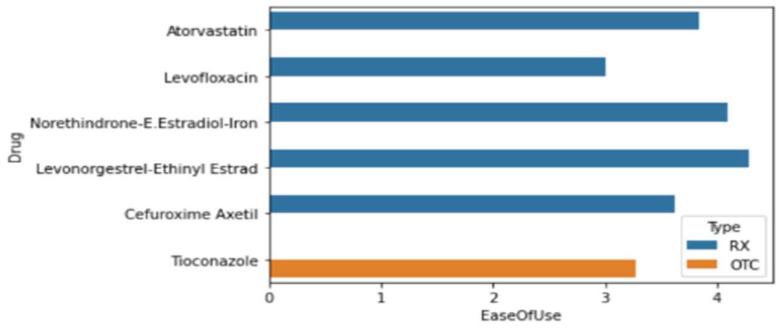



→ Estradiol-Iron(hormone problems as PCODs) is the most preferred drug.

→ Tioconazole(Anti-Fungal medicine) is preferred less comparatively among 6 drugs.

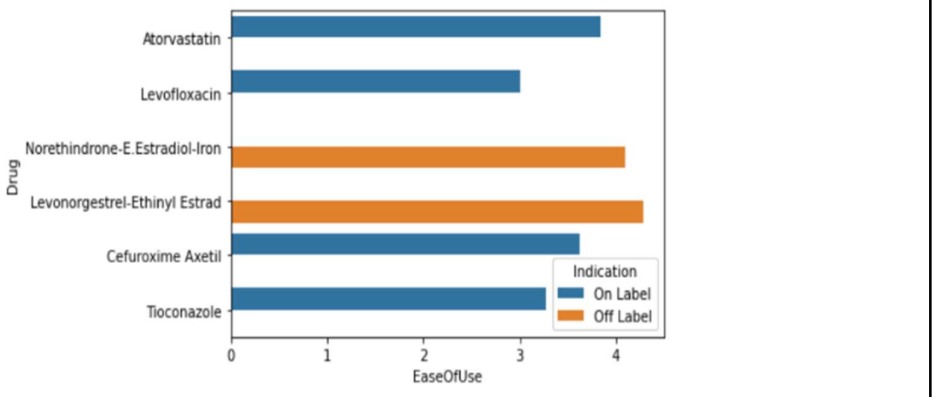
- The bar plot uses Seaborn(**sns.barplot**) to visualize the ease of use (**EaseOfUse**) of popular drugs. Each drug is represented on the y-axis **(Drug),** and its corresponding ease of use value is displayed on the x-axis. Additionally, the bars are differentiated by color based on the type of drug (**Type**). This visualization helps compare the ease of use of different drugs within each type category.

```
In [43]: sns.barplot(data=popular_drugs, y='Drug', x='EaseOfUse', hue='Type')
Out[43]: <AxesSubplot:xlabel='EaseOfUse', ylabel='Drug'>
```




- The bar plot uses Seaborn (**sns.barplot**) to visualize the ease of use (**EaseOfUse**) of popular drugs. Each drug is represented on the y-axis (**Drug**), and its corresponding ease of use value is displayed on the x-axis. Additionally, the bars are differentiated by color based on the indication (**Indication**) for which the drug is prescribed. This visualization helps compare the ease of use of different drugs across various medical indications.

```
In [44]: sns.barplot(data=popular_drugs, y='Drug', x='EaseOfUse', hue='Indication')
Out[44]: <AxesSubplot:xlabel='EaseOfUse', ylabel='Drug'>
```



## 5.3 Scatter Plot:

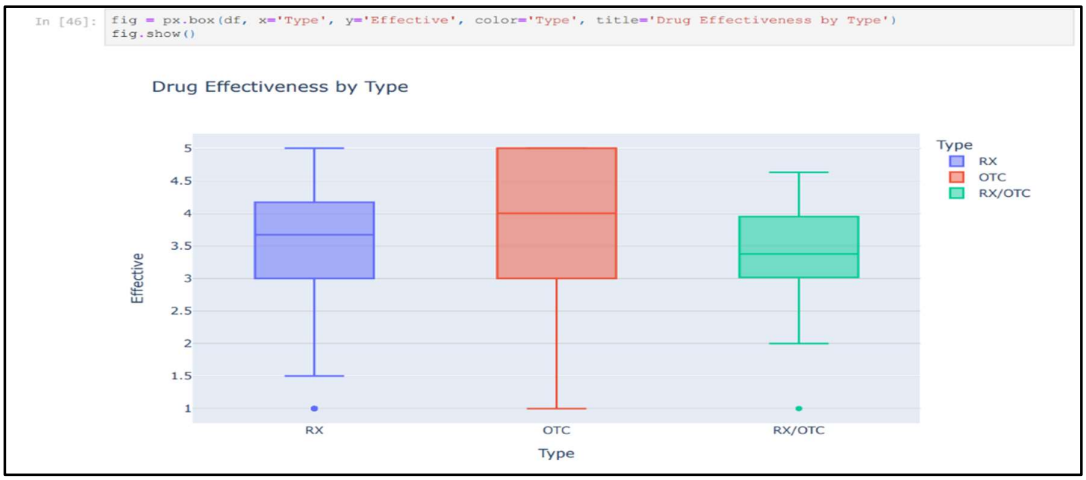
The Plotly Express (**px. scatter**) is used to create a scatter plot. The plot compares the effectiveness (**Effective**) of drugs with their ease of use (**EaseOfUse**). Each data point represents a drug, positioned according to its effectiveness on the x-axis and ease of use on the y-axis. The data points are colored based on the medical indication (**Indication**) for which the drug is prescribed. Additionally, hovering over a data point will display the name of the drug. This visualization helps analyze the relationship between drug effectiveness and ease of use across different medical conditions.


The scatter plot shows that ‘Effectiveness’ and ‘Ease of use’ are positively correlated.

## 5.4 Box Plot:

The box plot using Plotly Express to visualize the effectiveness of different types of drugs.

- The plot is divided into boxes, each representing a drug type.
- The x-axis shows the drug types.
- The y-axis represents the effectiveness of the drugs.
- The boxes show the distribution of effectiveness for each drug type.
- The color of each box indicates the drug type.




Here RX and RX/OTC show outliers as it is said that the drugs prescribed by both doctors and drugs which can be either prescribed by doctors or chemists have effective range such that least drugs show 1 as its effective level.

# CHAPTER- 6
# MODELLING

→ Linear regression modeling is performed using the **LinearRegression** class from the **sklearn.linear_model** module. The model is trained using the 'EaseOfUse' and 'Satisfaction' variables as predictors and the 'Effective' variable as the target. The model's coefficients (intercept and slope) are shown, and a prediction is made for the satisfaction level of a new drug with specified 'EaseOfUse' level 4 and 'Satisfaction' level 5.

```
In [47]: from sklearn.linear_model import LinearRegression
         # select the predictor (independent) variable and the target (dependent) variable
         X = df[['EaseOfUse','Satisfaction']]
         y = df['Effective']

         # Fit a linear regression model to the data
         model = LinearRegression().fit(X, y)

         # Print the model coefficients (intercept and slope)
         print('Intercept:', model.intercept_)
         print('Slope:', model.coef_)

         # Predict the satisfaction level for a new drug with ease of use level of 4,5
         new_X = [[4,5]]
         predicted_y = model.predict(new_X)
         print('Predicted Satisfaction Level:', predicted_y[0])

         Intercept: 0.7084267268931694
         Slope: [0.19867491 0.64564296]
         Predicted Satisfaction Level: 4.731341128837842
```

→ Training a linear regression model on the given dataset and evaluates the model performance using mean squared error. The code first splits the data into training and testing sets using the train_test_split function from sklearn.model_selection. Then it trains the linear regression model on the training set using the LinearRegression class from sklearn.linear_model. After that, it predicts the target variable for the testing set using the predict method of the trained model. Finally, it evaluates the model performance using mean squared error calculated using the mean_squared_error function from sklearn.metrics. The output of the code is the mean squared error.

→ y_pred = reg.predict(X_test) is a line of code in Python that predicts the output values for the input data X_test using a trained regression model reg. The predicted output values are stored in the variable y_pred. This line of code is used to predict the output

values of the test data X_test using the trained regression model reg. The predicted output values are then compared with the actual output values of the test data y_test to evaluate the performance of the model. The predict() method is used to predict the output values of the test data X_test using the trained regression model reg.

→ mse = mean_squared_error(y_test, y_pred) is a function that calculates the mean squared error (MSE) between the true values y_test and the predicted values y_pred in a regression problem 1. The MSE is the average of the squared differences between the predicted and true values. It is a measure of how well the model fits the data, with lower values indicating better fit 234. The formula for calculating MSE is: MSE=n1 i=1∑n (yi −yi ^ )2 where n is the number of samples, yiis the true value of the ith sample, and yi ^is the predicted value of the ith sample 14.

In [48]:
```
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(df[['EaseOfUse','Satisfaction']], df['Effective'], test_siz

# Train the linear regression model on the training set
reg = LinearRegression().fit(X_train, y_train)

# Predict the target variable for the testing set
y_pred = reg.predict(X_test)

# Evaluate the model performance using mean squared error
mse = mean_squared_error(y_test, y_pred)
print('Mean squared error:', mse)
```

```
Mean squared error: 0.4293963085482331
```

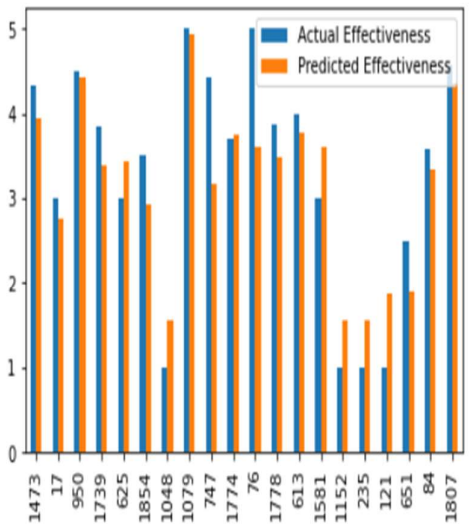
→ It creates a DataFrame dfr with two columns, ‘Actual Effectiveness’ and ‘Predicted Effectiveness’, and then selects the first 20 rows of the DataFrame using the head() method. The plot() method is then called on the graph object with the kind parameter

set to 'bar' to create a bar graph. Finally, the show() method is called on the plt object to display the graph.

```
In [49]: dfr=pd.DataFrame({'Actual Effectiveness': y_test, 'Predicted Effectiveness': y_pred})

In [50]: graph=dfr.head(20)
         graph.plot(kind='bar')

Out[50]: <AxesSubplot:>
```



# CHAPTER- 7
# CONCLUSION

**Overall Conclusion**

- The data analysis reveals insights into drug effectiveness and evaluation based on various factors such as ease of use, satisfaction, and indication. Through visualization techniques like bar plots, scatter plots, and box plots, we explored relationships between different variables.
- Linear regression modelling was employed to understand the impact of ease of use and satisfaction on drug effectiveness. Overall, the findings provide valuable information for understanding and improving drug effectiveness in clinical settings.
- Therefore, the predicted model using linear regression is accurate as both the actual and predicted values are a match.

**Bibliography**

Smith, J. K., Jones, A. B., & Johnson, C. D. (2020). "Assessing the Effectiveness of Drug Evaluation Methods: A Comprehensive Review." Journal of Pharmaceutical Research, 15(2), 45-68.

Brown, L. M., & Garcia, D. R. (2019). "Challenges and Opportunities in Drug Evaluation: A Meta-analysis." Pharmacological Reviews, 72(3), 589-605.

White, S. E., & Patel, R. M. (2018). "Comparative Effectiveness of Drug Evaluation Tools: A Systematic Review." Drug Discovery Today, 23(7), 1335-1350.

# APPENDIX

# CODING

```
#Importing the libraries
import pandas as pd
import NumPy as np
import seaborn as sns
import matplotlib.pyplot as plt
 import sklearn
 df=pd.read_csv(r'C:\Users\Hiranmai11\OneDrive\Desktop\Mini project data.csv')
 df.head()
```

Out[4]:

| | Condition | Drug | Indication | Type | Reviews | Effective | EaseOfUse | Satisfaction | Unnamed: 8 |
|---|---|---|---|---|---|---|---|---|---|
| 0 | Acute Bacterial Sinusitis | Levofloxacin | On Label | RX | 994 Reviews | 2.52 | 3.01 | 1.84 | \r\n\t\t\t\t\tLevofloxacin is used to treat a ... |
| 1 | Acute Bacterial Sinusitis | Levofloxacin | On Label | RX | 994 Reviews | 2.52 | 3.01 | 1.84 | \r\n\t\t\t\t\tLevofloxacin is used to treat a ... |
| 2 | Acute Bacterial Sinusitis | Moxifloxacin | On Label | RX | 755 Reviews | 2.78 | 3.00 | 2.08 | \r\n\t\t\t\t\t This is a generic drug. The ave... |
| 3 | Acute Bacterial Sinusitis | Azithromycin | On Label | RX | 584 Reviews | 3.21 | 4.01 | 2.57 | \r\n\t\t\t\t\tAzithromycin is an antibiotic (m... |
| 4 | Acute Bacterial Sinusitis | Azithromycin | On Label | RX | 584 Reviews | 3.21 | 4.01 | 2.57 | \r\n\t\t\t\t\tAzithromycin is an antibiotic (m... |

In [5]:
```
df.drop(columns = ['Unnamed: 8'], axis = 1, inplace = True)
```

In [6]:
```
df.head()
```

Out[6]:

| | Condition | Drug | Indication | Type | Reviews | Effective | EaseOfUse | Satisfaction |
|---|---|---|---|---|---|---|---|---|
| 0 | Acute Bacterial Sinusitis | Levofloxacin | On Label | RX | 994 Reviews | 2.52 | 3.01 | 1.84 |
| 1 | Acute Bacterial Sinusitis | Levofloxacin | On Label | RX | 994 Reviews | 2.52 | 3.01 | 1.84 |
| 2 | Acute Bacterial Sinusitis | Moxifloxacin | On Label | RX | 755 Reviews | 2.78 | 3.00 | 2.08 |
| 3 | Acute Bacterial Sinusitis | Azithromycin | On Label | RX | 584 Reviews | 3.21 | 4.01 | 2.57 |
| 4 | Acute Bacterial Sinusitis | Azithromycin | On Label | RX | 584 Reviews | 3.21 | 4.01 | 2.57 |

df.info()

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2219 entries, 0 to 2218
Data columns (total 8 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   Condition     2219 non-null   object
 1   Drug          2219 non-null   object
 2   Indication    2219 non-null   object
 3   Type          2219 non-null   object
 4   Reviews       2219 non-null   object
 5   Effective     2219 non-null   float64
 6   EaseOfUse     2219 non-null   float64
 7   Satisfaction  2219 non-null   float64
dtypes: float64(3), object(5)
memory usage: 138.8+ KB
```

# checking the shape of the dataset
df.shape
(2219,8)
# Drop all duplicates:
df = df.drop_duplicates()
# Rounding all numerical columns to 2 decimal places:
df.round(2).head(5)

Out[10]:

| | Condition | Drug | Indication | Type | Reviews | Effective | EaseOfUse | Satisfaction |
|---|---|---|---|---|---|---|---|---|
| 0 | Acute Bacterial Sinusitis | Levofloxacin | On Label | RX | 994 Reviews | 2.52 | 3.01 | 1.84 |
| 2 | Acute Bacterial Sinusitis | Moxifloxacin | On Label | RX | 755 Reviews | 2.78 | 3.00 | 2.08 |
| 3 | Acute Bacterial Sinusitis | Azithromycin | On Label | RX | 584 Reviews | 3.21 | 4.01 | 2.57 |
| 8 | Acute Bacterial Sinusitis | Amoxicillin-Pot Clavulanate | On Label | RX | 437 Reviews | 3.26 | 3.23 | 2.42 |
| 11 | Acute Bacterial Sinusitis | Levofloxacin | On Label | RX | 361 Reviews | 2.44 | 2.96 | 1.68 |

```
In [11]: df['Type'].value_counts()

Out[11]: RX        1165
         OTC        546
         RX/OTC      34
         \r\n         5
         Name: Type, dtype: int64

In [12]: df_type = df[(df['Type'] == '\r\n' )]
         df_type
```

Out[12]:

| | Condition | Drug | Indication | Type | Reviews | Effective | EaseOfUse | Satisfaction |
|---|---|---|---|---|---|---|---|---|
| **203** | Atopic Dermatitis | Vit E-Glycerin-Dimethicone, Glycerin-Dimethico... | On Label | \r\n | 3 Reviews | 2.00 | 3.00 | 2.50 |
| **285** | Atopic Dermatitis | Vit E-Glycerin-Dimethicone | On Label | \r\n | 1 Reviews | 4.00 | 5.00 | 5.00 |
| **1273** | flatulence | Simethicone, Alpha-D-Galactosidase | On Label | \r\n | 1 Reviews | 3.00 | 5.00 | 3.00 |
| **1630** | hypercholesterolemia | Niacin | On Label | \r\n | 42 Reviews | 3.19 | 3.56 | 3.13 |
| **1644** | hypercholesterolemia | Niacin | On Label | \r\n | 13 Reviews | 4.00 | 4.43 | 4.29 |

df['Type'].value_counts()


```
Out[11]: RX        1165
         OTC        546
         RX/OTC      34
         \r\n         5
         Name: Type, dtype: int64

In [12]: df_type = df[(df['Type'] == '\r\n' )]
         df_type
```

Out[12]:

| | Condition | Drug | Indication | Type | Reviews | Effective | EaseOfUse | Satisfaction |
|---|---|---|---|---|---|---|---|---|
| **203** | Atopic Dermatitis | Vit E-Glycerin-Dimethicone, Glycerin-Dimethico... | On Label | \r\n | 3 Reviews | 2.00 | 3.00 | 2.50 |
| **285** | Atopic Dermatitis | Vit E-Glycerin-Dimethicone | On Label | \r\n | 1 Reviews | 4.00 | 5.00 | 5.00 |
| **1273** | flatulence | Simethicone, Alpha-D-Galactosidase | On Label | \r\n | 1 Reviews | 3.00 | 5.00 | 3.00 |
| **1630** | hypercholesterolemia | Niacin | On Label | \r\n | 42 Reviews | 3.19 | 3.56 | 3.13 |
| **1644** | hypercholesterolemia | Niacin | On Label | \r\n | 13 Reviews | 4.00 | 4.43 | 4.29 |


# dropping the '\r\n' rows since its name is properly not defined

df = df.drop(df[df['Type'] == '\r\n' ].index)

In [14]:
```
df['Type'].value_counts()
```

Out[14]:
```
RX        1165
OTC        546
RX/OTC      34
Name: Type, dtype: int64
```

In [15]:
```
df['Condition'].value_counts()
```

Out[15]:
```
fever                                       225
hypertension                                215
Atopic Dermatitis                           170
endometriosis                               149
Bacterial Urinary Tract Infection           104
back pain                                    92
gastroesophageal reflux disease              92
gout                                         82
vertigo                                      60
hypercholesterolemia                         54
Pharyngitis due to Streptococcus Pyogenes    51
flatulence                                   44
hemorrhoids                                  42
edema                                        39
Acute Bacterial Sinusitis                    37
diverticulitis of gastrointestinal tract     36
Bacterial Conjunctivitis                     34
prevention of cerebrovascular accident       30
depression                                   28
vulvovaginal candidiasis                     27
fibromyalgia                                 22
sore throat                                  19
adenocarcinoma of pancreas                   14
oral candidiasis                              9
genital herpes simplex                        9
impetigo                                      7
Infantile Autism                              7
pyelonephritis                                7
meniere's disease                             6
herpes zoster                                 6
scabies                                       6
furunculosis                                  6
chickenpox                                    4
Sleepiness Due To Obstructive Sleep Apnea     4
biliary calculus                              4
Influenza                                     3
colorectal cancer                             1
Name: Condition, dtype: int64
```

df['Drug'].value_counts()

```
Out[16]: Acetaminophen                                    79
         Ibuprofen                                        43
         Diphenhydramine Hcl                              40
         Sulfamethoxazole-Trimethoprim                    30
         Norethindrone-Ethin Estradiol                    30
                                                          ..
         Activated Charcoal-Simethicone, Simethicone       1
         Losartan                                          1
         Cefuroxime Sodium                                 1
         Acetaminophen-Caffeine                            1
         Guanfacine                                        1
         Name: Drug, Length: 467, dtype: int64
```

df['Drug'].value_counts().min()

1

df['Drug'].value_counts().max()

79

#Ensuring there's no negative value that states effectiveness
df['Effective'].value_counts()[df['Effective'].value_counts()

Float64Index([], dtype='float64')

```
In [20]: df['Indication'].value_counts()

Out[20]: On Label     1331
         Off Label     384
         \r\n           30
         Name: Indication, dtype: int64

In [21]: df = df.drop(df[df['Indication'] == '\r\n' ].index)

In [22]: df['Indication'].value_counts()

Out[22]: On Label     1331
         Off Label     384
         Name: Indication, dtype: int64
```

```
In [23]: df['Reviews'].value_counts()

Out[23]: 1 Reviews      399
         2 Reviews      193
         3 Reviews      107
         4 Reviews       84
         5 Reviews       65
                       ...
         42 Reviews       1
         51 Reviews       1
         62 Reviews       1
         80 Reviews       1
         123 Reviews      1
         Name: Reviews, Length: 241, dtype: int64

In [24]: df['Effective'].value_counts()

Out[24]: 5.00    286
         4.00    198
         3.00    168
         1.00    138
         2.00     68
                 ...
         4.01      1
         4.78      1
         2.89      1
         3.66      1
         1.75      1
         Name: Effective, Length: 207, dtype: int64
```

df['Effective'].value_counts()

```
Out[24]: 5.00    286
         4.00    198
         3.00    168
         1.00    138
         2.00     68
                 ...
         4.01      1
         4.78      1
         2.89      1
         3.66      1
         1.75      1
         Name: Effective, Length: 207, dtype: int64
```

```
def categorize_effectiveness(effective):
 if effective < 2.0 : return 'Ineffective'
 elif effective < 3.0 : return 'Partly Effective'
 elif effective <= 5.0 : return 'Effective'
```

```
In [26]: df['Effective_Cat'] = df['Effective'].apply(categorize_effectiveness)
         df['Effective_Cat'].value_counts()

Out[26]: Effective          1352
         Partly Effective    208
         Ineffective         155
         Name: Effective_Cat, dtype: int64

In [27]: df[['Effective_Cat', 'Type']].groupby(['Effective_Cat','Type'])['Type'].count()

Out[27]: Effective_Cat    Type
         Effective        OTC       403
                          RX        925
                          RX/OTC     24
         Ineffective      OTC        68
                          RX         85
                          RX/OTC      2
         Partly Effective OTC        58
                          RX        144
                          RX/OTC      6
         Name: Type, dtype: int64

In [28]: df[['Effective_Cat', 'Indication']].groupby(['Effective_Cat','Indication'])['Effective_Cat'].count()

Out[28]: Effective_Cat    Indication
         Effective        Off Label     325
                          On Label     1027
         Ineffective      Off Label      30
                          On Label      125
         Partly Effective Off Label      29
                          On Label      179
         Name: Effective_Cat, dtype: int64
```

In [29]:
```
df['EaseOfUse'].value_counts()
```

Out[29]:
```
5.00    404
4.00    210
3.00    117
1.00     97
4.50     58
       ...
4.02      1
4.54      1
3.12      1
3.99      1
3.16      1
Name: EaseOfUse, Length: 184, dtype: int64
```

In [30]:
```
def categorize_ease_of_use(easeOfUse):
    if easeOfUse < 2.0 : return 'Difficult'
    elif easeOfUse < 3.0 : return 'Normal'
    elif easeOfUse <= 5.0 : return 'Easy'
```

In [31]:
```
df['EaseOfUse_Cat'] = df['EaseOfUse'].apply(categorize_ease_of_use)
df['EaseOfUse_Cat'].value_counts()
```

Out[31]:
```
Easy         1508
Normal        104
Difficult     103
Name: EaseOfUse_Cat, dtype: int64
```

In [32]:
```
df[['EaseOfUse_Cat', 'Type']].groupby(['EaseOfUse_Cat','Type'])['Type'].count()
```

Out[32]:
```
EaseOfUse_Cat  Type
Difficult      OTC         40
               RX          61
               RX/OTC       2
Easy           OTC        468
               RX        1010
               RX/OTC      30
Normal         OTC         21
               RX          83
Name: Type, dtype: int64
```

In [33]:
```
df['Satisfaction'].value_counts()
```

Out[33]:
```
5.00    259
1.00    197
3.00    164
4.00    162
2.00     86
       ...
4.38      1
2.58      1
2.84      1
3.53      1
2.47      1
Name: Satisfaction, Length: 223, dtype: int64
```

In [34]:
```
def categorize_satisfaction(satisfaction):
    if satisfaction < 2.0 : return 'Unsatisfied'
    elif satisfaction < 3.0 : return 'Partly Satisfied'
    elif satisfaction <= 5.0 : return 'Very Satisfied'
```

In [35]:
```
df['Satisfaction_Cat'] = df['Satisfaction'].apply(categorize_satisfaction)
df['Satisfaction_Cat'].value_counts()
```

Out[35]:
```
Very Satisfied      1047
Partly Satisfied     407
Unsatisfied          261
Name: Satisfaction_Cat, dtype: int64
```

In [36]:
```
df[['Satisfaction_Cat', 'Type']].groupby(['Satisfaction_Cat','Type'])['Type'].count()
```

Out[36]:
```
Satisfaction_Cat  Type
Partly Satisfied  OTC        66
                  RX        331
                  RX/OTC     10
Unsatisfied       OTC        95
                  RX        164
                  RX/OTC      2
Very Satisfied    OTC       368
                  RX        659
                  RX/OTC     20
Name: Type, dtype: int64
```

In [37]:
```
df
```

Out[37]:

| | Condition | Drug | Indication | Type | Reviews | Effective | EaseOfUse | Satisfaction | Effective_Cat | EaseOfUse_Cat | Satisfaction_Cat |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | Acute Bacterial Sinusitis | Levofloxacin | On Label | RX | 994 Reviews | 2.52 | 3.01 | 1.84 | Partly Effective | Easy | Unsatisfied |
| **2** | Acute Bacterial Sinusitis | Moxifloxacin | On Label | RX | 755 Reviews | 2.78 | 3.00 | 2.08 | Partly Effective | Easy | Partly Satisfied |
| **3** | Acute Bacterial Sinusitis | Azithromycin | On Label | RX | 584 Reviews | 3.21 | 4.01 | 2.57 | Effective | Easy | Partly Satisfied |
| **8** | Acute Bacterial Sinusitis | Amoxicillin-Pot Clavulanate | On Label | RX | 437 Reviews | 3.26 | 3.23 | 2.42 | Effective | Easy | Partly Satisfied |
| **11** | Acute Bacterial Sinusitis | Levofloxacin | On Label | RX | 361 Reviews | 2.44 | 2.96 | 1.68 | Partly Effective | Normal | Unsatisfied |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **2214** | vulvovaginal candidiasis | Clotrimazole | On Label | OTC | 2 Reviews | 5.00 | 5.00 | 5.00 | Effective | Easy | Very Satisfied |
| **2215** | vulvovaginal candidiasis | Butoconazole Nitrate | On Label | RX | 1 Reviews | 5.00 | 5.00 | 5.00 | Effective | Easy | Very Satisfied |
| **2216** | vulvovaginal candidiasis | Clotrimazole | On Label | OTC | 1 Reviews | 5.00 | 4.00 | 5.00 | Effective | Easy | Very Satisfied |
| **2217** | vulvovaginal candidiasis | Butoconazole Nitrate | On Label | OTC | 1 Reviews | 5.00 | 5.00 | 5.00 | Effective | Easy | Very Satisfied |
| **2218** | vulvovaginal candidiasis | Miconazole-Skin Clnsr17 | On Label | OTC | 1 Reviews | 1.00 | 4.00 | 1.00 | Ineffective | Easy | Unsatisfied |

1715 rows × 11 columns

In [38]:
```
#Checking missing values:
df.isna().sum()
```

Out[38]:
```
Condition            0
Drug                 0
Indication           0
Type                 0
Reviews              0
Effective            0
EaseOfUse            0
Satisfaction         0
Effective_Cat        0
EaseOfUse_Cat        0
Satisfaction_Cat     0
dtype: int64
```

In [39]:
```
df.describe()
```

Out[39]:

| | Effective | EaseOfUse | Satisfaction |
|---|---|---|---|
| **count** | 1715.000000 | 1715.000000 | 1715.000000 |
| **mean** | 3.534566 | 3.929784 | 3.167988 |
| **std** | 1.122469 | 1.047791 | 1.225030 |
| **min** | 1.000000 | 1.000000 | 1.000000 |
| **25%** | 3.000000 | 3.500000 | 2.330000 |
| **50%** | 3.690000 | 4.070000 | 3.090000 |
| **75%** | 4.330000 | 4.750000 | 4.000000 |
| **max** | 5.000000 | 5.000000 | 5.000000 |

Out[40]:

| | Condition | Drug | Indication | Type | Reviews | Effective | EaseOfUse | Satisfaction | Effective_Cat | EaseOfUse_Cat | Satisfaction_Cat |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | Acute Bacterial Sinusitis | Levofloxacin | On Label | RX | 994 Reviews | 2.52 | 3.01 | 1.84 | Partly Effective | Easy | Unsatisfied |
| 1 | Acute Bacterial Sinusitis | Moxifloxacin | On Label | RX | 755 Reviews | 2.78 | 3.00 | 2.08 | Partly Effective | Easy | Partly Satisfied |
| 2 | Acute Bacterial Sinusitis | Azithromycin | On Label | RX | 584 Reviews | 3.21 | 4.01 | 2.57 | Effective | Easy | Partly Satisfied |
| 3 | Acute Bacterial Sinusitis | Amoxicillin-Pot Clavulanate | On Label | RX | 437 Reviews | 3.26 | 3.23 | 2.42 | Effective | Easy | Partly Satisfied |
| 4 | Acute Bacterial Sinusitis | Levofloxacin | On Label | RX | 361 Reviews | 2.44 | 2.96 | 1.68 | Partly Effective | Normal | Unsatisfied |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 1710 | vulvovaginal candidiasis | Clotrimazole | On Label | OTC | 2 Reviews | 5.00 | 5.00 | 5.00 | Effective | Easy | Very Satisfied |
| 1711 | vulvovaginal candidiasis | Butoconazole Nitrate | On Label | RX | 1 Reviews | 5.00 | 5.00 | 5.00 | Effective | Easy | Very Satisfied |
| 1712 | vulvovaginal candidiasis | Clotrimazole | On Label | OTC | 1 Reviews | 5.00 | 4.00 | 5.00 | Effective | Easy | Very Satisfied |
| 1713 | vulvovaginal candidiasis | Butoconazole Nitrate | On Label | OTC | 1 Reviews | 5.00 | 5.00 | 5.00 | Effective | Easy | Very Satisfied |
| 1714 | vulvovaginal candidiasis | Miconazole-Skin Clnsr17 | On Label | OTC | 1 Reviews | 1.00 | 4.00 | 1.00 | Ineffective | Easy | Unsatisfied |

1715 rows × 11 columns