Drugs Data Analysis

This presentation focuses on **drug data analysis**, highlighting key insights such as **most used drugs, common side effects, drug classes, pregnancy categories, and more**. The goal is to understand trends in **medications, side effects, and interactions** using **data visualization**.

-- BY Priya Rajurkar

Project Overview

- This project analyzes a dataset containing information about various drugs, their medical uses, side effects, classifications, and interactions. By examining these attributes, we can gain valuable insights into which drugs are most commonly used, their associated risks, and how they are categorized based on prescription type, pregnancy safety, and controlled substance status.
- The analysis involves visualizing key patterns using **bar charts, pie charts, scatter plots and steam graphs**. These visualizations help in understanding drug usage trends, highlighting the most prescribed drug classes, and identifying potential risks such as alcohol interactions or high-risk side effects. This project is useful for healthcare professionals, researchers, and consumers seeking data-driven insights into pharmaceuticals.

Technology Used

1 Python Programming

- Used for **data cleaning, processing, and analysis**.
- Helps in handling large datasets efficiently.

2 Pandas

- Used for **data manipulation and transformation**.
- Helps in filtering, grouping, and summarizing drug-related information.

3 Matplotlib & Seaborn

- **Matplotlib**: Used for creating detailed visualizations like **bar charts, scatter plots, and step graphs**.
- **Seaborn**: Used for aesthetically appealing **categorical and statistical visualizations** such as count plots and heatmaps.

4 Data Visualization Techniques

- **Bar Charts, Pie Charts, Scatter Plots, Step Graphs, and Steam Graphs** to highlight trends and patterns in drug usage.

Step 1 – Read The Data

```
df=pd.read_csv("drugs_side_effects_drugs_com.csv")
```

```
df
```

	drug_name	medical_condition	side_effects	generic_name	drug_classes	brand_names	activity	rx_otc	pregnancy_categ
0	doxycycline	Acne	(hives, difficult breathing, swelling in your ...	doxycycline	Miscellaneous antimalarials, Tetracydines	Acticlate, Adoxa CK, Adoxa Pak, Adoxa TT, Alod...	87%	Rx	
1	spironolactone	Acne	hives ; difficulty breathing; swelling of your...	spironolactone	Aldosterone receptor antagonists, Potassium-sp...	Aldactone, CaroSpir	82%	Rx	
2	minocycline	Acne	skin rash, fever, swollen glands, flu-like sym...	minocycline	Tetracydines	Dynacin, Minocin, Minolira, Solodyn, Ximino, V...	48%	Rx	
			problems						

Step 2 – Checking Data Types

```
df.dtypes
```

```
drug_name                     object
medical_condition             object
side_effects                  object
generic_name                  object
drug_classes                  object
brand_names                   object
activity                      object
rx_otc                        object
pregnancy_category            object
csa                           object
alcohol                       object
related_drugs                 object
medical_condition_description object
rating                       float64
no_of_reviews                float64
drug_link                     object
medical_condition_url         object
dtype: object
```

Step 3 – Finding Missing Value

```
df.isnull()
```

	drug_name	medical_condition	side_effects	generic_name	drug_classes	brand_names	activity	rx_otc	pregnancy_category	c
0	False	False	False	False	False	False	False	False	False	Fa
1	False	False	False	False	False	False	False	False	False	Fa
2	False	False	False	False	False	False	False	False	False	Fa
3	False	False	False	False	False	True	False	False	False	Fa
4	False	False	False	False	False	False	False	False	False	Fa
...	...	...	...	...	...	...	...	...	...	
2926	False	False	False	False	False	False	False	False	False	Fa
2927	False	False	False	False	False	False	False	False	False	Fa
2928	False	False	True	False	False	True	False	False	False	Fa
2929	False	False	False	False	False	True	False	False	True	Fa
2930	False	False	False	False	False	False	False	False	True	Fa

```
df.isnull().sum()
```

```
drug_name                     0
medical_condition             0
side_effects                124
generic_name                 43
drug_classes                 82
brand_names                1213
activity                      0
rx_otc                        1
pregnancy_category          229
csa                           0
alcohol                    1554
related_drugs              1469
medical_condition_description    0
rating                     1345
no_of_reviews              1345
drug_link                     0
medical_condition_url         0
dtype: int64
```

Step 4 – Handling Missing Value

```
df["rating"].fillna(0, inplace=True)
```

```
df["no_of_reviews"].fillna(0, inplace= True)
```

```
df["alcohol"].fillna("ND",inplace=True)
```

```
df["related_drugs"].fillna("ND",inplace=True)
```

```
df["pregnancy_category"].fillna("N",inplace=True)
```

```
df["brand_names"].fillna("UnKnown",inplace= True)
```

```
df["generic_name"].fillna("ND", inplace = True)
```

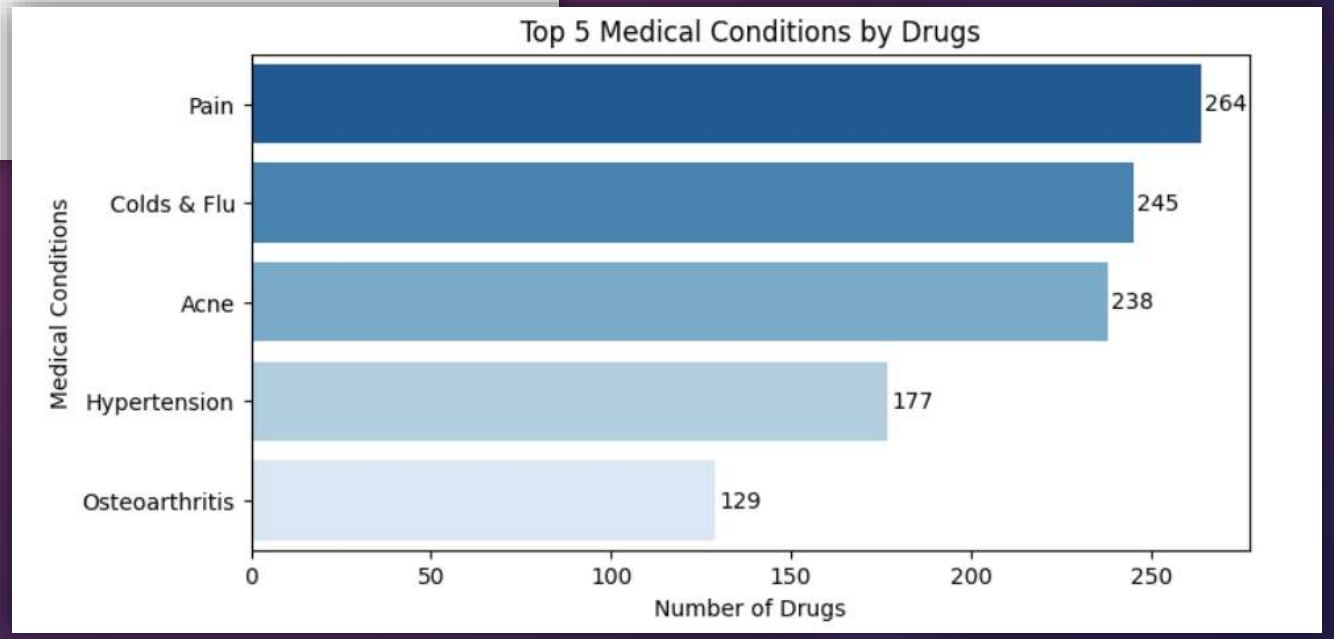
```
df["drug_classes"].fillna("ND",inplace = True)
```

```
df.dropna(subset=['rx_otc'], inplace=True)
```

Step 5 – Insights

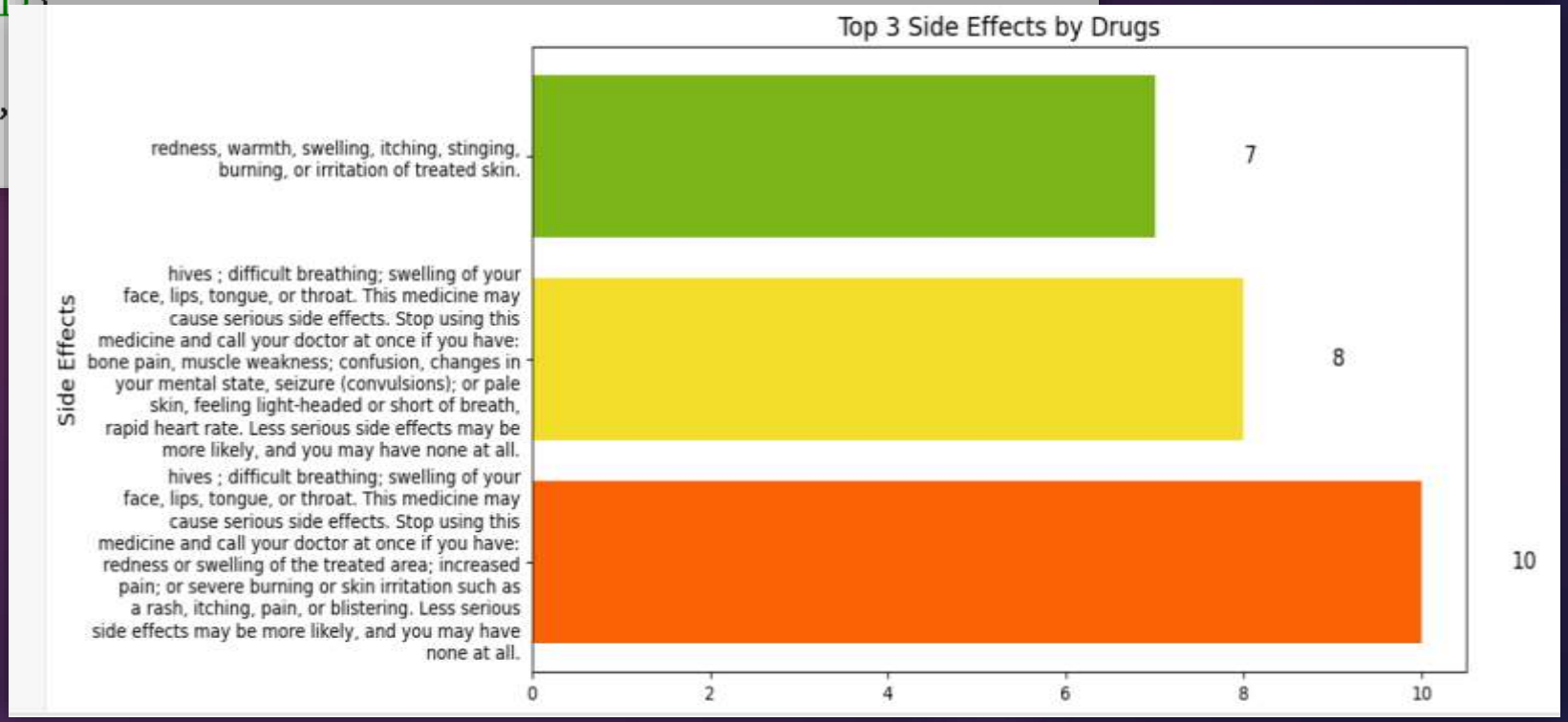
1] Top 5 Medical Conditions by Drugs

```
top_condition=df["medical_condition"].value_counts().nlargest(5)
plt.figure(figsize=(8, 4))
sns.barplot(x=top_condition.values, y=top_condition.index, palette="Blues_r")
for index, value in enumerate(top_condition.values):
    plt.text(value + 1, index, str(value), va='center', fontsize=10)
plt.xlabel("Number of Drugs")
plt.ylabel("Medical Conditions")
plt.title("Top 5 Medical Conditions by Drugs")
plt.show()
```



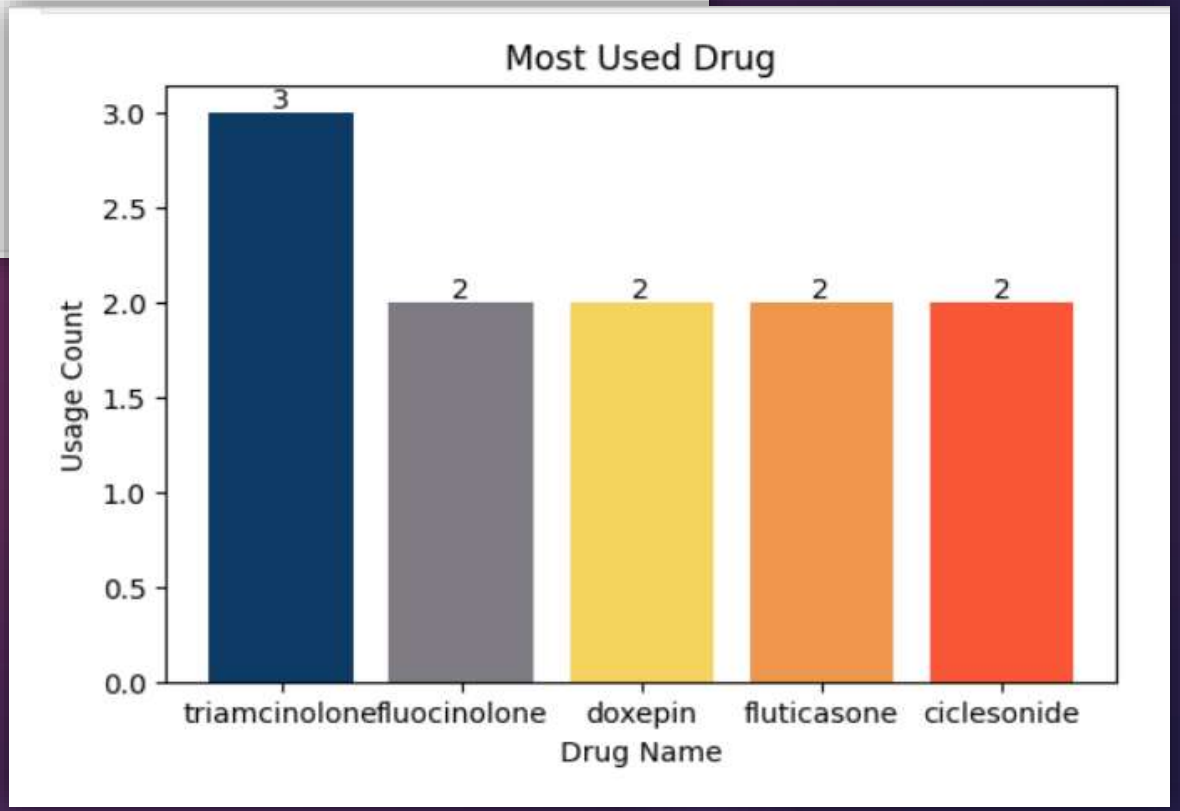
2] Top 3 Side Effects by Drugs

```
import textwrap
top_side_effects = df["side_effects"].value_counts().nlargest(3)
c = ["#FB6107", "#F3DE2C", "#7CB518"]
wrapped_labels = ['\n'.join(textwrap.wrap(label, width=50)) for label in top_side_effects.index]
plt.figure(figsize=(10, 6))
bars = plt.barh(wrapped_labels, top_side_effects.values, color=c)
for bar, value in zip(bars, top_side_effects.values):
    plt.text(value + 1, bar.get_y() + bar.get_height()/2, str(value), va='center', fontsize=12)
plt.xlabel("Number of Drugs", fontsize=12)
plt.ylabel("Side Effects", fontsize=12)
plt.title("Top 3 Side Effects by Drugs",
plt.show()
```



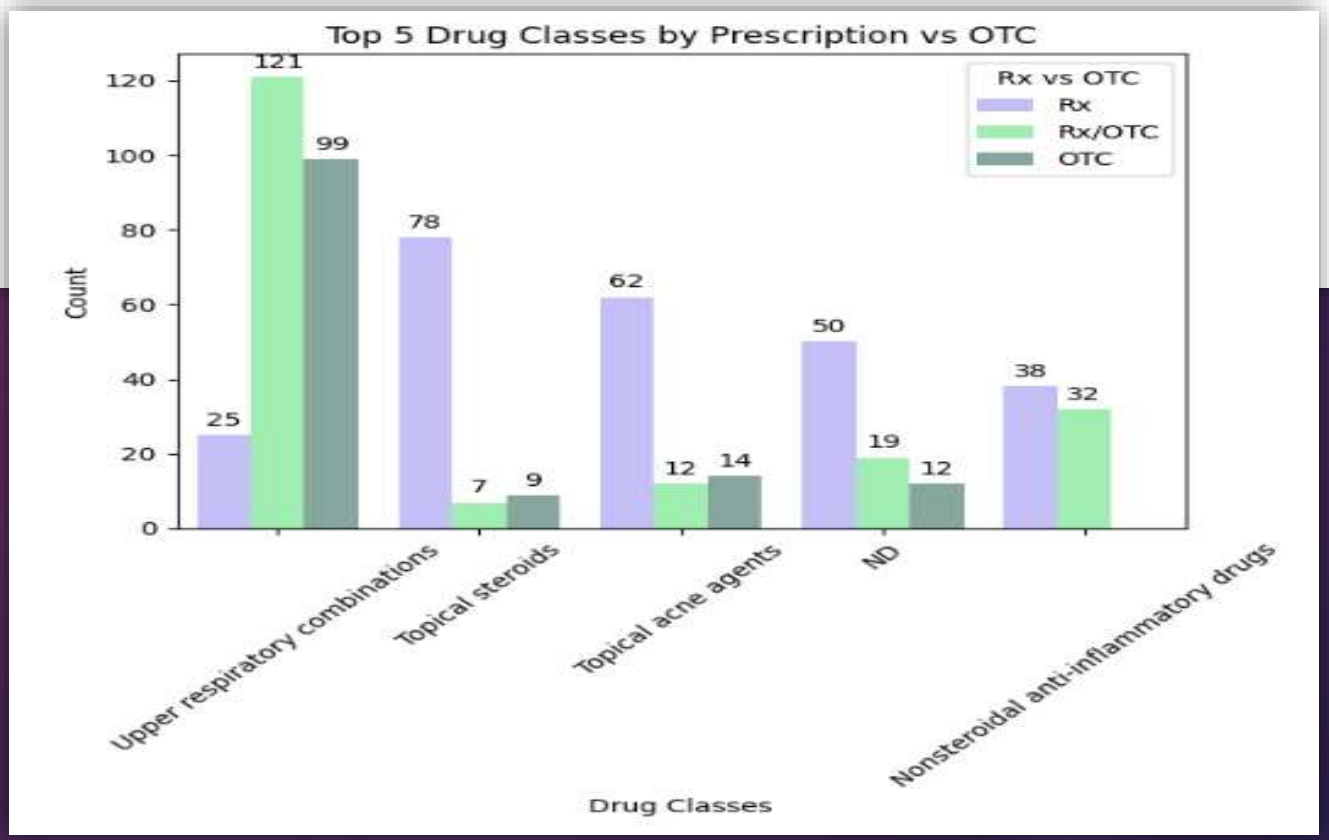
Top 3 Most used Drugs

```
most_used_drug = df['drug_name'].value_counts().nlargest(5)
c = ["#0D3B66", "#7F7B82", "#F4D35E", "#EE964B", "#F95738"]
plt.figure(figsize=(6, 4))
bars = plt.bar(most_used_drug.index, most_used_drug.values, color=c)
for bar, value in zip(bars, most_used_drug.values):
    plt.text(bar.get_x() + bar.get_width()/2, bar.get_height() - 0.001, str(value),
             ha='center', va='bottom', fontsize=10)
plt.xlabel("Drug Name")
plt.ylabel("Usage Count")
plt.title("Most Used Drug")
plt.show()
```



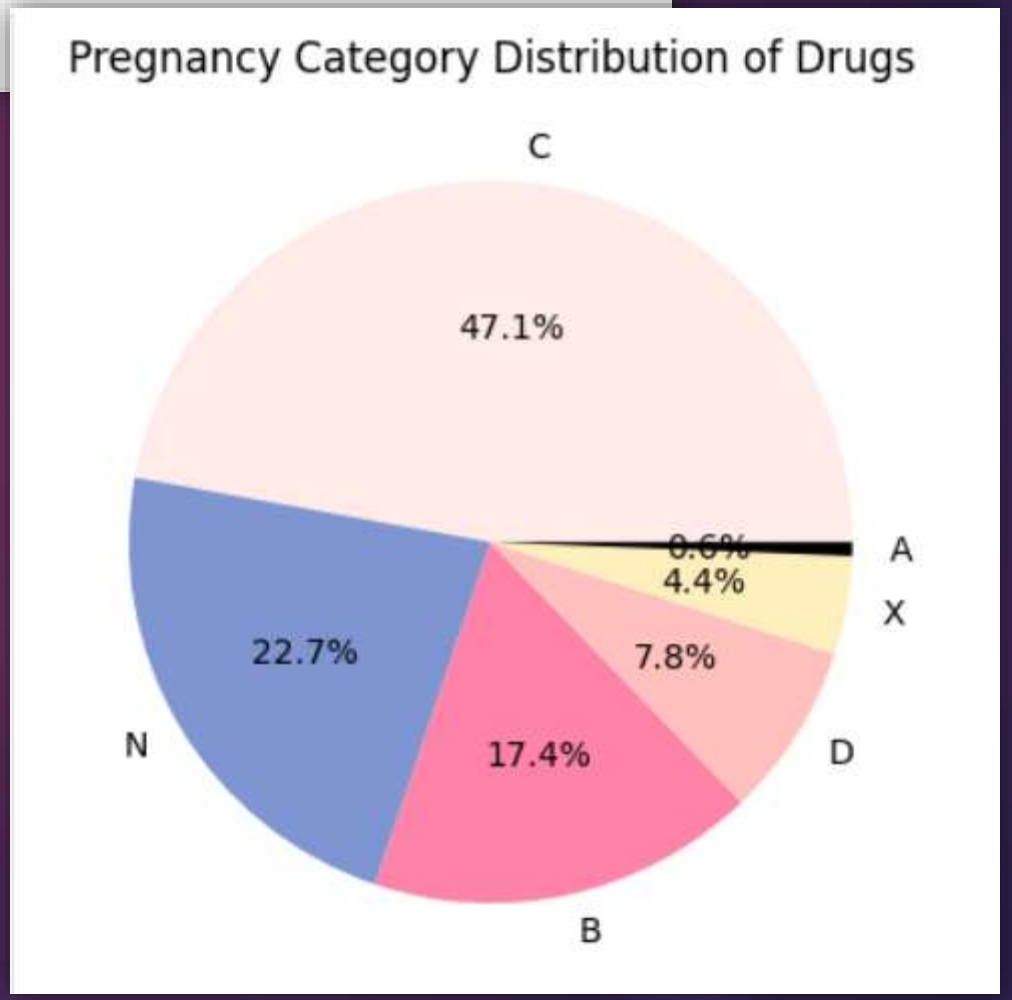
5] Top 5 Drug Classes by Rx vs OTC

```
top_5_classes = df['drug_classes'].value_counts().nlargest(5).index
c=["#BCB6FF","#94FBAB","#82ABA1"]
ax = sns.countplot(data=df[df['drug_classes'].isin(top_5_classes)], x="drug_classes", hue="rx_otc", order=top_5_classes,
                   palette= c)
for container in ax.containers:
    ax.bar_label(container, fmt='%d', label_type='edge', fontsize=10, padding=3)
plt.title("Top 5 Drug Classes by Prescription vs OTC")
plt.xlabel("Drug Classes")
plt.ylabel("Count")
plt.xticks(rotation=45)
plt.legend(title="Rx vs OTC")
plt.show()
```



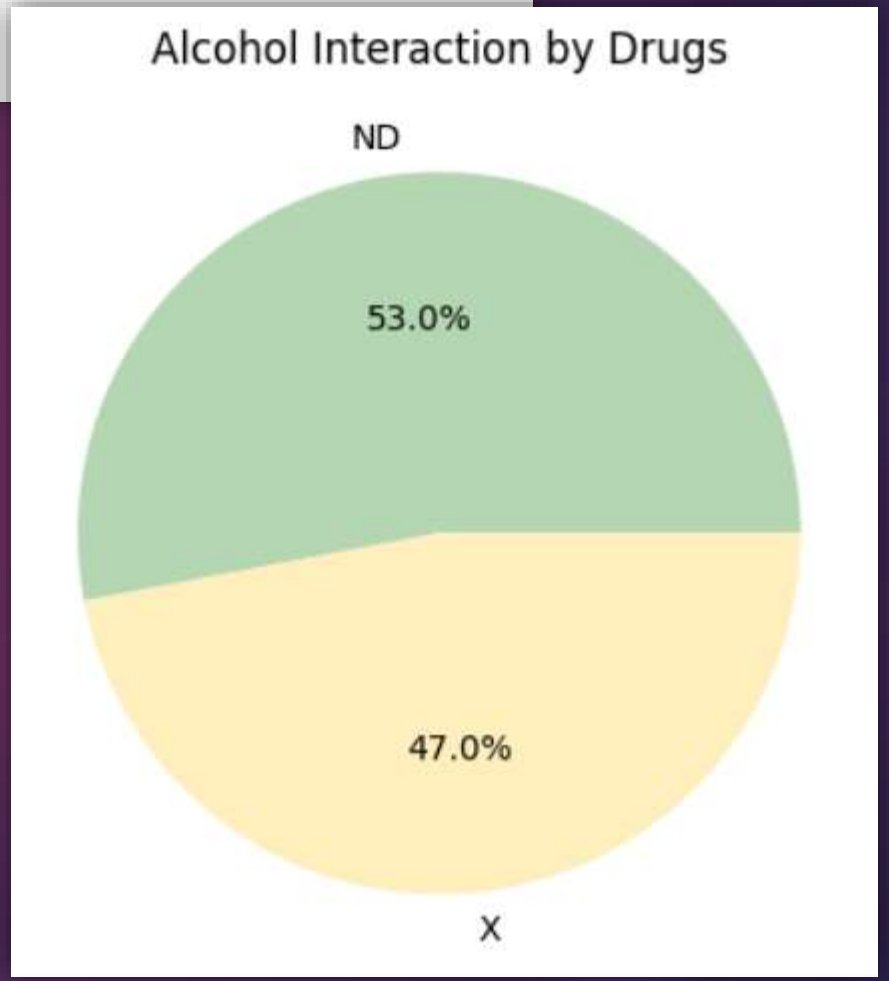
6] Pregnancy Category by Drugs

```
pregnancy_counts = df['pregnancy_category'].value_counts()
c=["#FFEBE7","#7F95D1","#FF82A9","#FFC0BE","#FFEFBD","#000000"]
plt.pie(pregnancy_counts, labels=pregnancy_counts.index, autopct='%1.1f%%', colors= c)
plt.title("Pregnancy Category Distribution of Drugs")
plt.show()
```



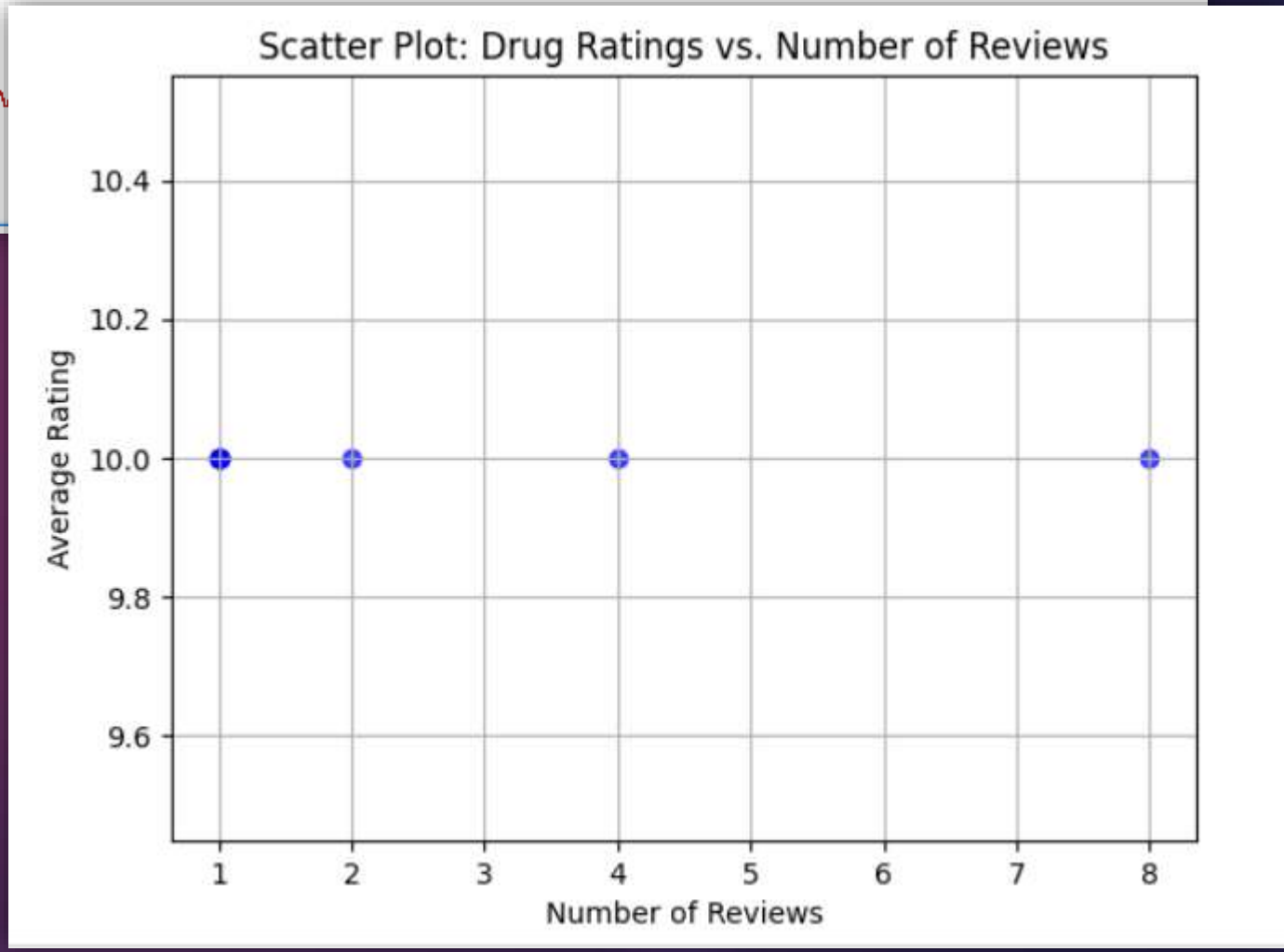
6]Alcohol Interaction by Drugs

```
alcohol_counts = df['alcohol'].value_counts()
c=["#B5D6B2","#FFEFBD"]
plt.pie(alcohol_counts, labels=alcohol_counts.index, autopct='%1.1f%%', colors = c)
plt.title("Alcohol Interaction by Drugs")
plt.show()
```



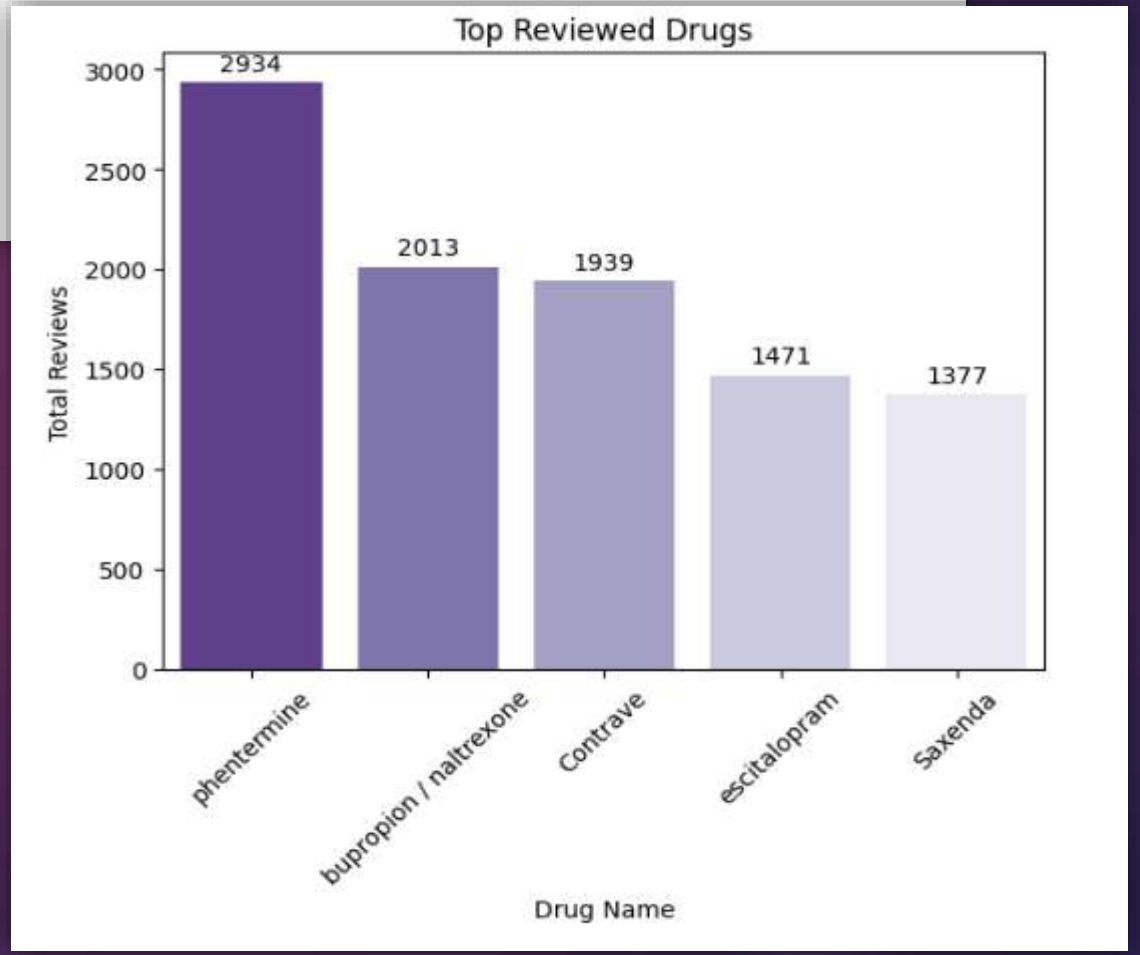
7] Top Rated Drugs

```
top_rated_drugs = df.groupby('drug_name').agg({'rating': 'mean', 'no_of_reviews': 'sum'}).nlargest(10, 'rating')
plt.scatter(top_rated_drugs['no_of_reviews'], top_rated_drugs['rating'], color='blue', alpha=0.7)
plt.xlabel("Number of Reviews")
plt.ylabel("Average Rating")
plt.title("Scatter Plot: Drug Ratings vs. Number of Rev
plt.grid(True)
plt.show()
```



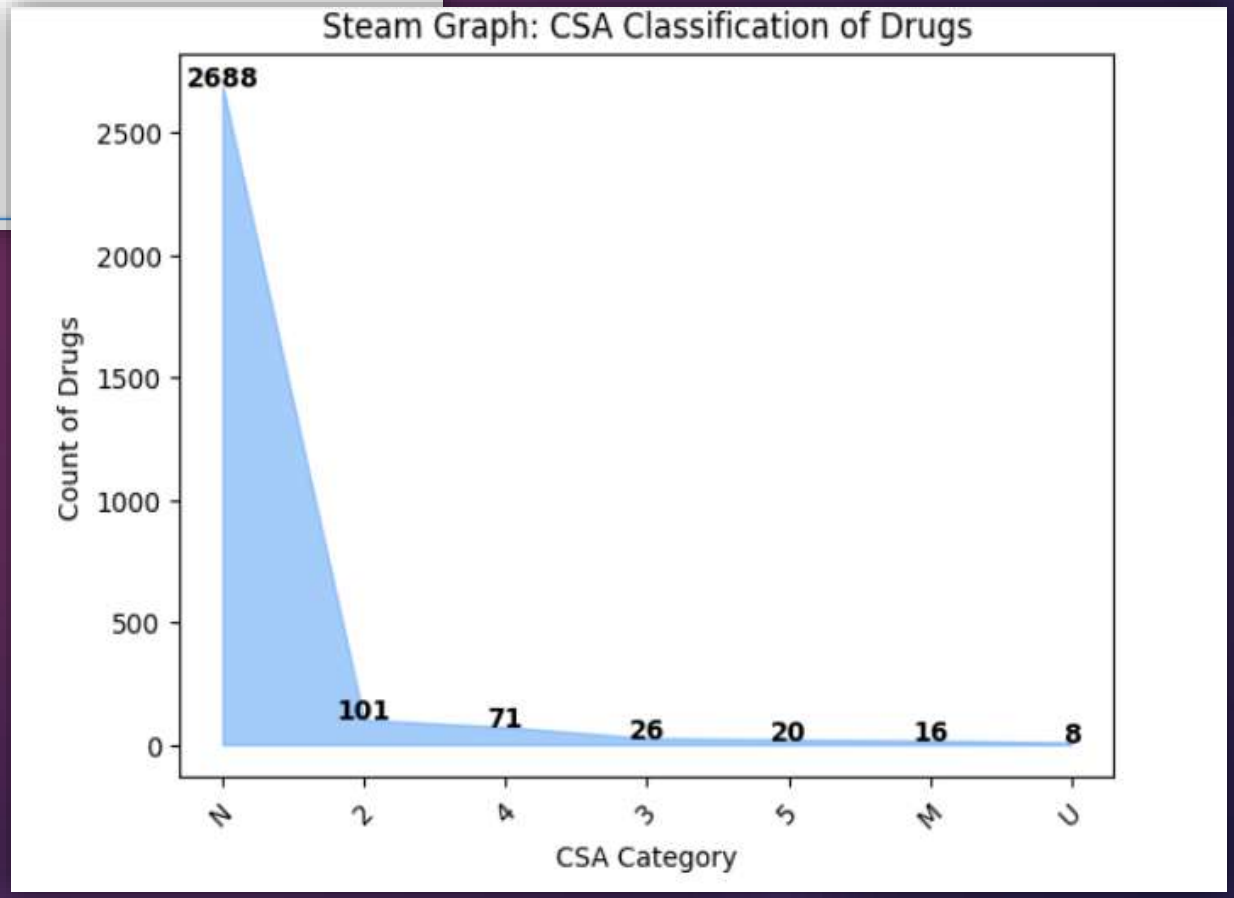
8] Top Reviewed Drugs

```
top_reviewed_drugs = df.groupby('drug_name')['no_of_reviews'].sum().nlargest(5)
ax= sns.barplot(x=top_reviewed_drugs.index, y=top_reviewed_drugs.values, palette="Purples_r")
for container in ax.containers:
    ax.bar_label(container, fmt='%d', label_type='edge', fontsize=10, padding=3)
plt.xlabel("Drug Name")
plt.xticks(rotation=45)
plt.ylabel("Total Reviews")
plt.title("Top Reviewed Drugs")
plt.show()
```



9] CSA category distribution.

```
import numpy as np
csa_counts = df['csa'].value_counts()
x = np.arange(len(csa_counts))
y = csa_counts.values
plt.fill_between(x, y, color="#67AAF9", alpha=0.6)
for i, value in enumerate(y):
    plt.text(i, value + 1, str(value), ha='center', fontsize=10, fontweight='bold')
plt.xticks(x, csa_counts.index, rotation=45)
plt.xlabel("CSA Category")
plt.ylabel("Count of Drugs")
plt.title("Steam Graph: CSA Classification of Drugs")
plt.show()
```



Summary

- This project provided valuable insights into drug usage, classifications, and their effects using **data analysis and visualization**. By analyzing a dataset containing details about **medical conditions, drug classes, side effects, and interactions**, we uncovered key trends that can help healthcare professionals and consumers make informed decisions.
- The analysis revealed the **top medical conditions with the highest number of available drugs**, the **most common side effects**, and the **most frequently used drugs**. Additionally, the study highlighted **drug classifications based on prescription type (Rx vs. OTC), pregnancy safety categories, and alcohol interactions**. By leveraging **bar charts, pie charts, scatter plots, and step graphs**, we visualized these insights effectively.
- Overall, this project demonstrated the **power of data-driven decision-making in the pharmaceutical industry**, enabling better understanding and awareness of drug usage trends.

Thank you for your time and attention!

This analysis provided valuable insights into drug classifications, side effects, and interactions. We hope this information helps in making informed decisions about pharmaceuticals.