250k Medicines Usage, Side Effects, and Substitute Analysis Report

Opeyemi Fayipe

17th September, 2024

Introduction

This project focuses on analyzing a dataset containing information on over 248,000 drugs. The key objectives are:

- To understand the distribution of side effects among drugs.
- To explore the availability of substitutes.
- To analyze the habit-forming potential of drugs across various therapeutic classes.

The insights from this analysis aim to inform healthcare professionals and pharmaceutical stakeholders about drug safety and potential risks.

Data Description

The dataset consists of various columns representing the drugs' names, side effects, therapeutic classes, chemical composition, and whether they are habit-forming. The main columns include:

1. **Drug Name**: The commercial name of the drug.
2. **Substitute**: Alternative drugs with similar compositions.
3. **Side Effects:** Information on adverse reactions.
4. **Therapeutic Class:** It is classifying medical drugs according to their functions. Each therapeutic class is a group of similar medications classified together because they are intended to treat the same medical conditions. For instance, respiratory is one of the classes, so all medications treating this illness are in the same class.
5. **Action Class:** It is the way of classifying medications based on actions they perform such as "H2 Receptor Blocker" It Block H2 receptors in parietal cells of the stomach → decreases gastric acid secretion. So drugs with similar action are grouped under "H2 Receptor Blocker".
6. **Chemical Class**: As the name suggests, it is grouped based on the chemical compound used.
7. **Habit Forming:** Indicates whether the drug is habit-forming.

Data Loading and Initial Exploration

Load the Dataset, to get a basic overview

```
import pandas as pd

# Load the dataset
file_path = (r"C:\Users\Admin\OneDrive\Desktop\Unified Mentor Projects\250k Medicines Usage, Side Effects and Substitutes.csv")
df = pd.read_csv(file_path)
df.head()
```

[15]:

	id	name	substitute0	substitute1	substitute2	substitute3	substitute4	sideEffect0	sideEffect1	sideEffect2	...	use3	use4	Chemical Class	Habit Forming
0	1	augmentin 625 duo tablet	Penciclav 500 mg/125 mg Tablet	Moxikind-CV 625 Tablet	Moxiforce-CV 625 Tablet	Fightox 625 Tablet	Novamox CV 625mg Tablet	Vomiting	Nausea	Diarrhea	...	Not specified	Not specified	Unknown	NO
1	2	azithral 500 tablet	Zithrocare 500mg Tablet	Azax 500 Tablet	Zady 500 Tablet	Cazithro 500mg Tablet	Trulimax 500mg Tablet	Vomiting	Nausea	Abdominal pain	...	Not specified	Not specified	Macrolides	NO
2	3	ascoril ls syrup	Solvin LS Syrup	Ambrodil-LX Syrup	Zerotuss XP Syrup	Capex LS Syrup	Broxum LS Syrup	Nausea	Vomiting	Diarrhea	...	Not specified	Not specified	Unknown	NO
3	4	allegra 120mg tablet	Lcfex Tablet	Etofex 120mg Tablet	Nexofex 120mg Tablet	Fexise 120mg Tablet	Histafree 120 Tablet	Headache	Drowsiness	Dizziness	...	Not specified	Not specified	Diphenylmethane Derivative	NO
4	5	avil 25 tablet	Eralet 25mg Tablet	No substitute available	No substitute available	No substitute available	No substitute available	Sleepiness	Dryness in mouth	No known side effects	...	Not specified	Not specified	Pyridines Derivatives	NO

5 rows × 62 columns

```
# Display basic info about the dataset
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 248218 entries, 0 to 248217
Data columns (total 62 columns):
 #   Column                          Non-Null Count   Dtype
---  ------                          --------------   -----
 0   id                              248218 non-null  int64
 1   name                            248218 non-null  object
 2   substitute0                     248218 non-null  object
 3   substitute1                     248218 non-null  object
 4   substitute2                     248218 non-null  object
 5   substitute3                     248218 non-null  object
 6   substitute4                     248218 non-null  object
 7   sideEffect0                     248218 non-null  object
 8   sideEffect1                     248218 non-null  object
 9   sideEffect2                     248218 non-null  object
 10  sideEffect3                     248218 non-null  object
 11  sideEffect4                     248218 non-null  object
 12  sideEffect5                     248218 non-null  object
 13  sideEffect6                     248218 non-null  object
 14  sideEffect7                     248218 non-null  object
 15  sideEffect8                     248218 non-null  object
 16  sideEffect9                     248218 non-null  object
 17  sideEffect10                    248218 non-null  object
 18  sideEffect11                    248218 non-null  object
 19  sideEffect12                    248218 non-null  object
 20  sideEffect13                    248218 non-null  object
 21  sideEffect14                    248218 non-null  object
 22  sideEffect15                    248218 non-null  object
 23  sideEffect16                    248218 non-null  object
 24  sideEffect17                    248218 non-null  object
 25  sideEffect18                    248218 non-null  object
 26  sideEffect19                    248218 non-null  object
```

```
# Display basic info about the dataset
df.describe()
```

	id
count	248218.000000
mean	124109.500000
std	71654.508896
min	1.000000
25%	62055.250000
50%	124109.500000
75%	186163.750000
max	248218.000000

Data Cleaning and Preparation

The dataset contained missing values, particularly in the Chemical Class, Therapeutic Class, and Action Class columns. Missing data was handled as follows:

```
# Check for missing values
print(df.isnull().sum())
```

```
id                      0
name                    0
substitute0          9597
substitute1         14351
substitute2         17985
substitute3         21362
substitute4         24256
sideEffect0             0
sideEffect1          9802
sideEffect2         18718
sideEffect3         40580
sideEffect4         84658
sideEffect5        116960
sideEffect6        156361
sideEffect7        180468
sideEffect8        199712
sideEffect9        210510
sideEffect10       220944
sideEffect11       227887
sideEffect12       231936
sideEffect13       233491
sideEffect14       237799
sideEffect15       240537
sideEffect16       242209
sideEffect17       242836
sideEffect18       243703
```

Handling Missing Values

- **Substitute Columns:** Missing values were replaced with 'No substitute available'.
- **Side Effects Columns:** Missing values were replaced with 'No known side effects'.
- **Habit Forming Column:** Missing values were filled with 'NO'.
- Chemical, Therapeutic, and Action Classes: Missing values were imputed with 'Unknown'.

```
# Fill missing values for substitutes and side effects
substitute_cols = [f'substitute{i}' for i in range(5)]
side_effect_cols = [f'sideEffect{i}' for i in range(42)]
df[substitute_cols] = df[substitute_cols].fillna('No substitute available')
df[side_effect_cols] = df[side_effect_cols].fillna('No known side effects')

# Fill missing values in 'Habit Forming' column
df['Habit Forming'] = df['Habit Forming'].fillna('NO')

# Fill missing usage columns with 'Not specified'
usage_cols = [f'use{i}' for i in range(5)]
df[usage_cols] = df[usage_cols].fillna('Not specified')

# Handle missing values in 'Chemical Class', 'Therapeutic Class', and
'Action Class' with 'Unknown'
df['Chemical Class'] = df['Chemical Class'].fillna('Unknown')
```

```
df['Therapeutic Class'] = df['Therapeutic Class'].fillna('Unknown')
df['Action Class'] = df['Action Class'].fillna('Unknown')

# Verify that there are no more missing values in these columns
missing_values_summary = df[['Chemical Class', 'Therapeutic Class', 'Action
Class']].isnull().sum()
print(missing_values_summary)

# Check if missing values are handled properly
missing_summary = df.isnull().sum()
missing_summary
```

```
Chemical Class       0
Therapeutic Class    0
Action Class         0
dtype: int64
```

```
id                   0
name                 0
substitute0          0
substitute1          0
substitute2          0
substitute3          0
substitute4          0
sideEffect0          0
sideEffect1          0
sideEffect2          0
sideEffect3          0
sideEffect4          0
sideEffect5          0
sideEffect6          0
sideEffect7          0
sideEffect8          0
sideEffect9          0
sideEffect10         0
sideEffect11         0
```

Exploratory Data Analysis (EDA)

Python Code Snippet:

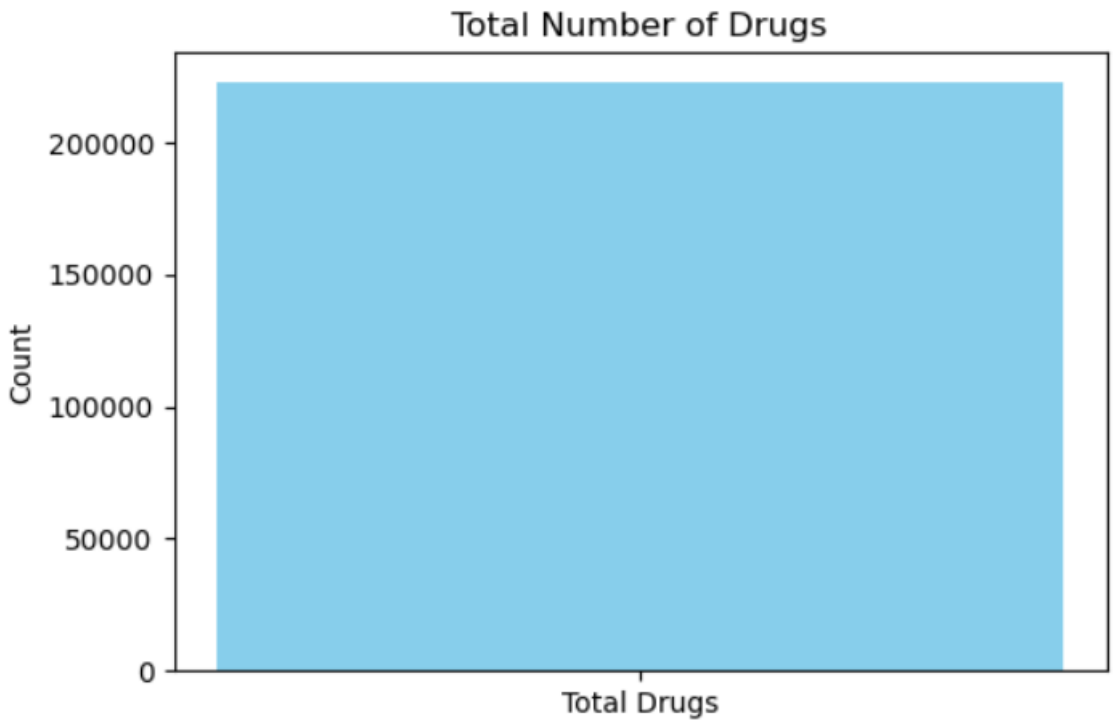
```
# Total number of unique drugs
total_drugs = df['name'].nunique()
print(f"Total number of drugs: {total_drugs}")

import matplotlib.pyplot as plt
plt.figure(figsize=(6,4))
plt.bar(['Total Drugs'], [total_drugs], color='skyblue')
plt.title('Total Number of Drugs')
plt.ylabel('Count')
plt.show()
```

Total Number of Drugs

- There are 222,825 unique drugs in the dataset.

Visualization:

```
Total number of drugs: 222825
```

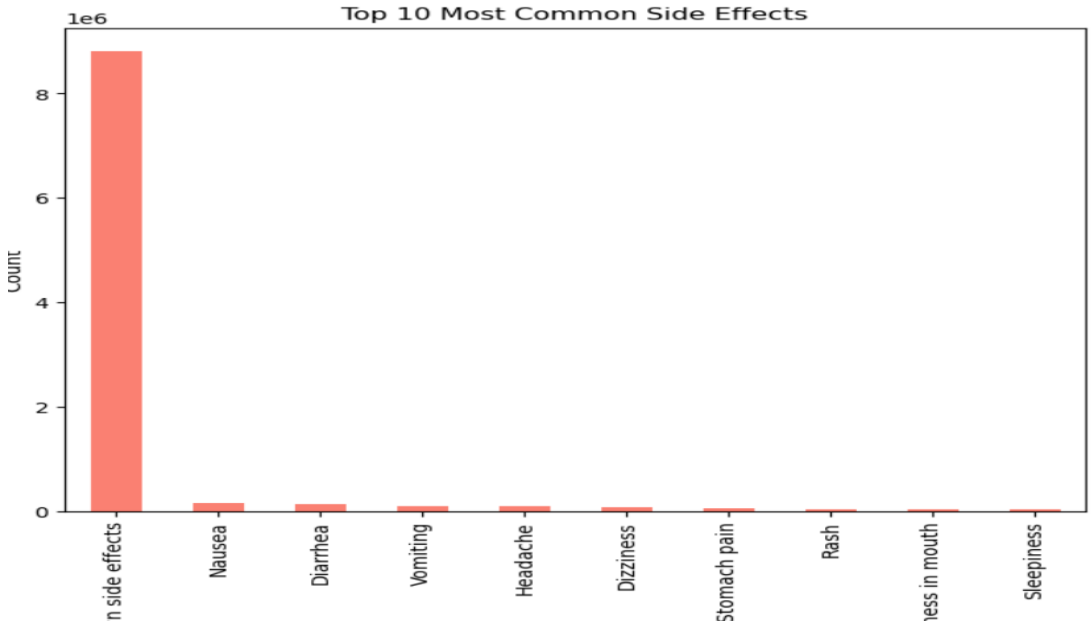


Most Common Side Effects

Python Code Snippet:

```
# Combining all side effect columns
side_effect_columns = [col for col in df.columns if 'sideEffect' in col]
side_effects = pd.Series(df[side_effect_columns].values.ravel()).dropna()

# Getting the top 10 most common side effects
common_side_effects = side_effects.value_counts().head(10)

# Plotting the most common side effects
plt.figure(figsize=(8,6))
common_side_effects.plot(kind='bar', color='salmon')
plt.title('Top 10 Most Common Side Effects')
plt.xlabel('Side Effect')
plt.ylabel('Count')
plt.xticks()
plt.show()
```

Visualization:
The top 10 most common side effects are **Nausea**, **Vomiting**, and **Headache**, among others. With the highest being “No known side effects”.

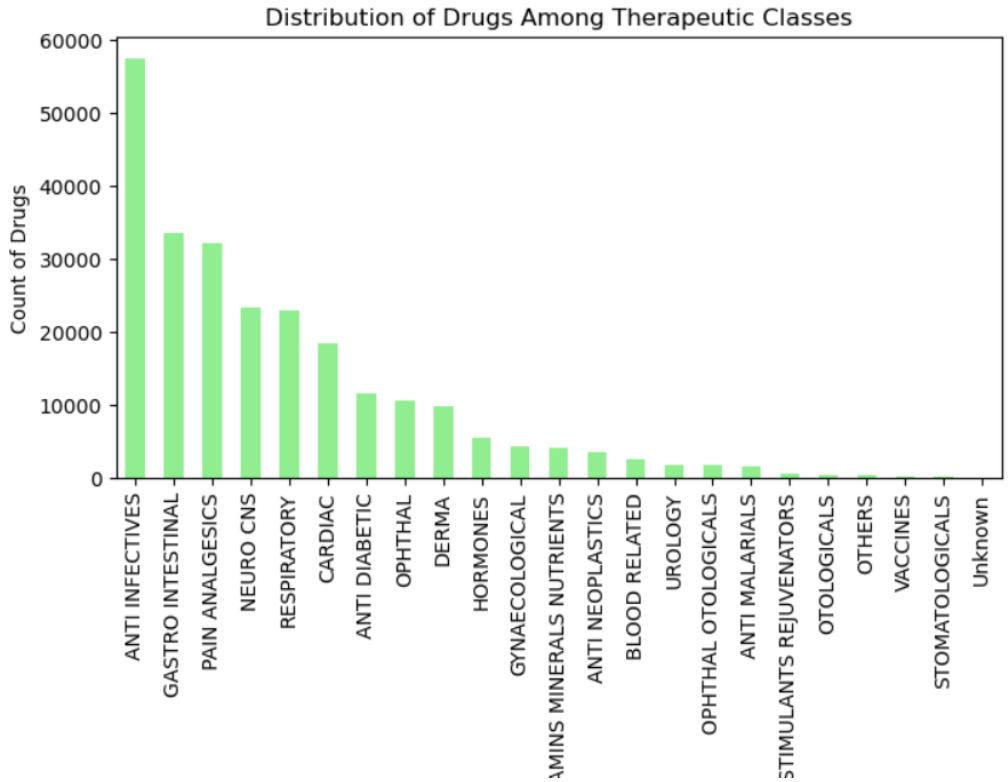


Distribution of Drugs Among Therapeutic Classes

Python Code Snippet:

```
# Distribution of drugs among therapeutic classes
# Distribution of drugs among therapeutic classes
therapeutic_class_distribution = df['Therapeutic Class'].value_counts()

# Plotting the distribution
plt.figure(figsize=(12,8))
therapeutic_class_distribution.plot(kind='bar', color='lightgreen')
plt.title('Distribution of Drugs Among Therapeutic Classes')
plt.xlabel('Therapeutic Class')
plt.ylabel('Count of Drugs')
plt.xticks(rotation=90)
plt.show()
```

Visualization:
The majority of drugs belong to the Anti Infectives, Gastrointestinal, and Respiratory therapeutic classes.



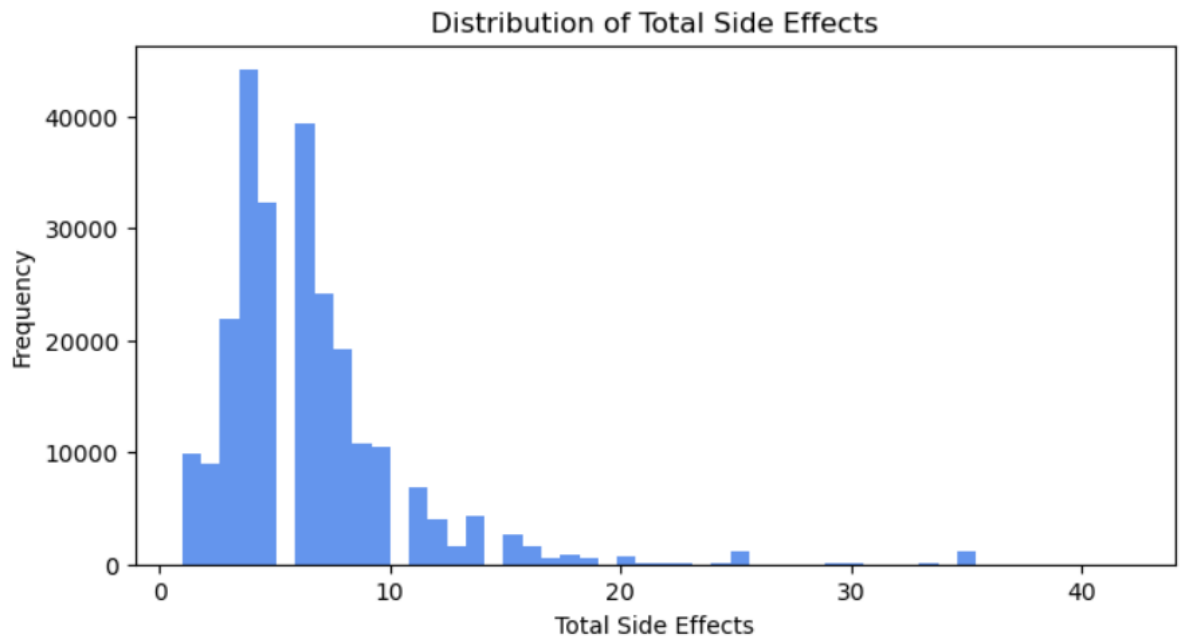
Total Side Effects

The dataset shows a wide range of side effects per drug, with most drugs having fewer than 10 side effects.

Python Code Snippet:

```
# List of side effect columns
side_effect_cols = [col for col in df.columns if 'sideEffect' in col]

# Creating the 'side_effect_count' column by counting non-'No known side
effects' entries
df['side_effect_count'] = df[side_effect_cols].apply(lambda row: row[row !=
'No known side effects'].count(), axis=1)

# Distribution of total side effects
plt.figure(figsize=(8,4))
plt.hist(df['side_effect_count'], bins=50, color='cornflowerblue')
plt.title('Distribution of Total Side Effects')
plt.xlabel('Total Side Effects')
plt.ylabel('Frequency')
plt.show()
```

Visualization:



Habit-Forming Drugs Analysis

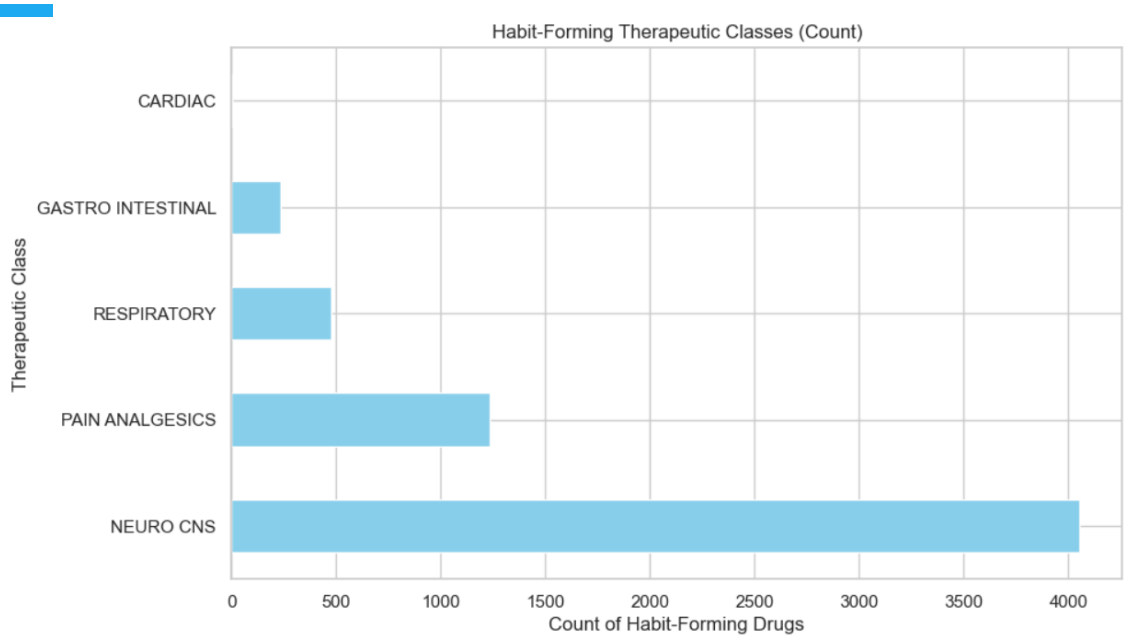
Habit-Forming Drugs by Therapeutic Class

The Neuro CNS and Pain Analgesics therapeutic classes have the highest number of habit-forming drugs, with 4054 and 1233 drugs, respectively.

Therapeutic Class	Values
NEURO CNS	4054
PAIN ANALGESICS	1233
RESPIRATORY	474
GASTROINTESTINAL	236
CARDIAC	6

Python Code Snippet:

```
# Cross-checking the count of habit-forming drugs per therapeutic class
habit_forming_classes = df[df['Habit Forming'] == 'YES']['Therapeutic
Class'].value_counts()
print(habit_forming_classes)


# Plot for Habit-Forming drugs
plt.figure(figsize=(10, 6))
habit_forming_classes.plot(kind='barh', color='skyblue')
plt.title('Habit-Forming Therapeutic Classes (Count)')
plt.xlabel('Count of Habit-Forming Drugs')
plt.ylabel('Therapeutic Class')
plt.show()
```



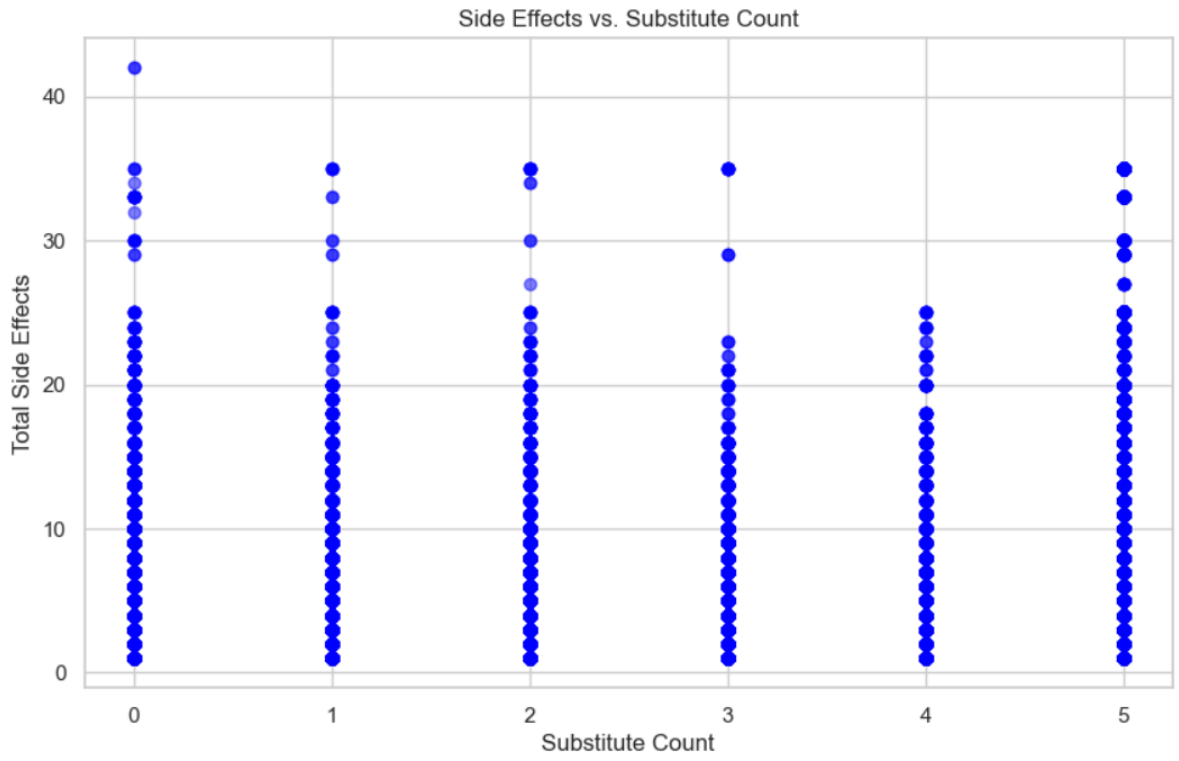
Substitutes and Side Effects

Substitutes and Side Effects Relationship

An analysis of the relationship between the number of substitutes and side effects suggests that there is minimal correlation between these two factors.

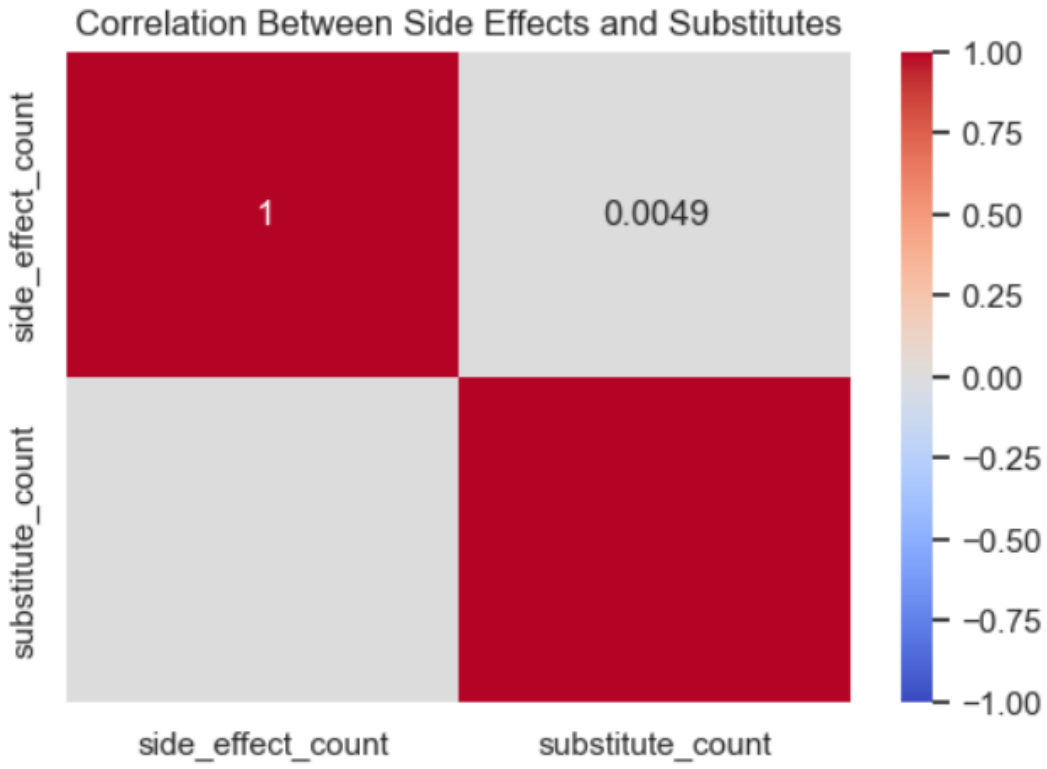
Python Code Snippets

```
# Relationship between side effect count and substitutes
plt.figure(figsize=(10,6))
plt.scatter(df['substitute_count'], df['side_effect_count'], alpha=0.5,
color='blue')
plt.title('Side Effects vs. Substitute Count')
plt.xlabel('Substitute Count')
plt.ylabel('Total Side Effects')
plt.show()
```




```
# Calculate the correlation between numerical columns
correlation_matrix = df[['side_effect_count', 'substitute_count']].corr()

# Display the correlation matrix
print(correlation_matrix)

# Plot the correlation heatmap
plt.figure(figsize=(6, 4))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', vmin=-1,
vmax=1)
plt.title('Correlation Between Side Effects and Substitutes')
plt.show()
```

	side_effect_count	substitute_count
side_effect_count	1.000000	0.004866
substitute_count	0.004866	1.000000



Habit-Forming Drugs with Substitutes

About **5756 habit-forming drugs** have substitutes, while **247 habit-forming drugs** do not.

Code Snippet:

```
# Habit-forming drugs with and without substitutes
habit_with_substitutes = df[(df['Habit Forming'] == 'YES') &
(df['substitute_count'] > 0)].shape[0]
habit_without_substitutes = df[(df['Habit Forming'] == 'YES') &
(df['substitute_count'] == 0)].shape[0]

print(f"Habit-forming drugs with substitutes: {habit_with_substitutes}")
print(f"Habit-forming drugs without substitutes:
{habit_without_substitutes}")
```

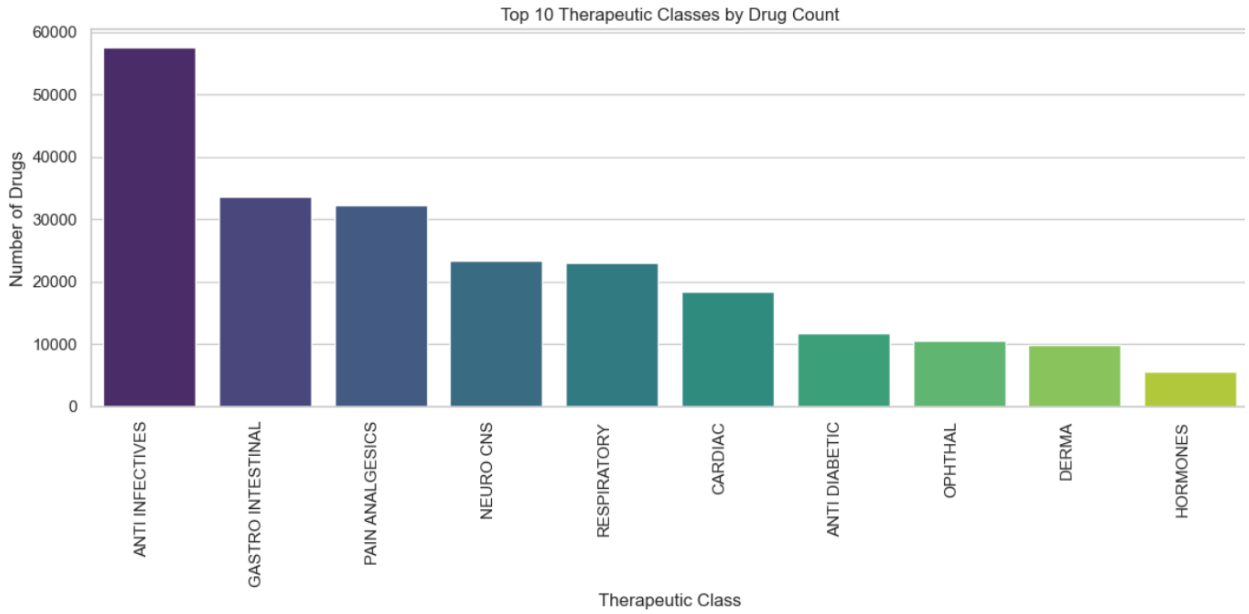
```
Habit-forming drugs with substitutes: 5756
Habit-forming drugs without substitutes: 247
```

Top 10 Therapeutic Classes by Drug Count

Python Code Snippet:

```
import seaborn as sns

# Set plot styles for better visuals
sns.set(style="whitegrid")
plt.figure(figsize=(12, 6))

# Step 1: Distribution of drugs by Therapeutic Class
therapeutic_class_counts = df['Therapeutic Class'].value_counts().head(10)  # Top
10 therapeutic classes
sns.barplot(x=therapeutic_class_counts.index, y=therapeutic_class_counts.values,
palette="viridis")
plt.title('Top 10 Therapeutic Classes by Drug Count')
plt.ylabel('Number of Drugs')
plt.xlabel('Therapeutic Class')
plt.xticks(rotation=90, ha='right')
plt.tight_layout()
plt.show()
```

Visualization:

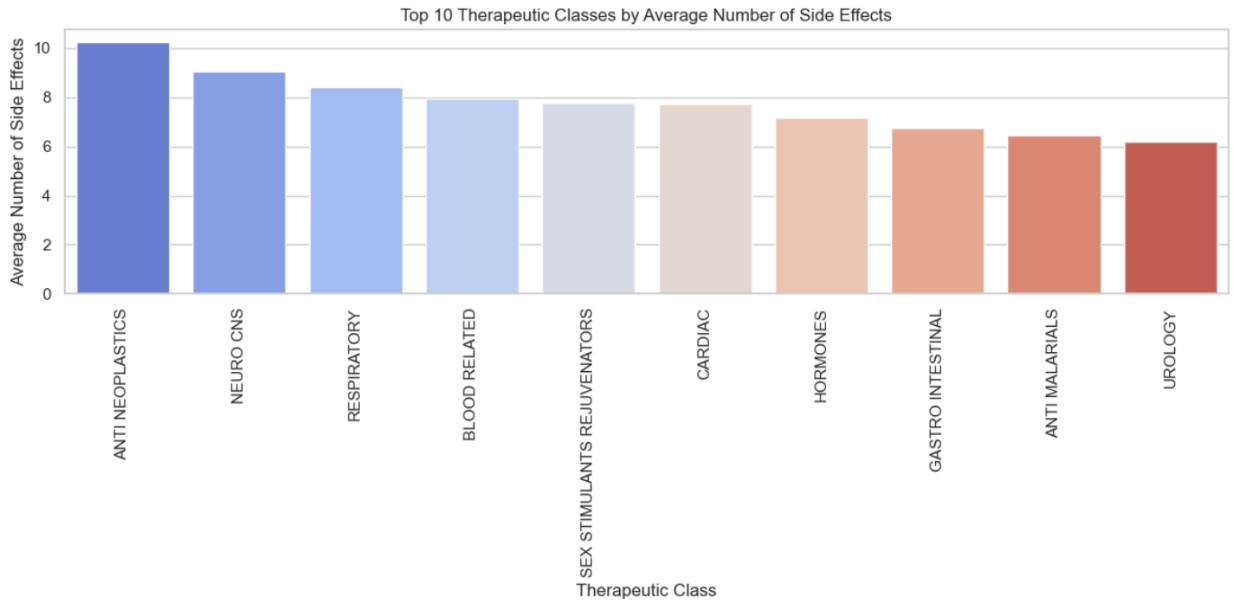


Top 10 Therapeutic Classes by Average Number of Side Effects

Python Code Snippets:

```
# Analyzing Average Side Effects Per Therapeutic Class
# Count the number of non-empty side effects for each drug
df['side_effect_count'] = df[side_effect_cols].apply(lambda row: row[row != 'No known side effects'].count(), axis=1)

# Calculate the average side effects per therapeutic class
avg_side_effects_per_class = df.groupby('Therapeutic Class')['side_effect_count'].mean().sort_values(ascending=False)

# Plot the result
plt.figure(figsize=(12, 6))
sns.barplot(x=avg_side_effects_per_class.index[:10], y=avg_side_effects_per_class.values[:10], palette="coolwarm")
plt.title('Top 10 Therapeutic Classes by Average Number of Side Effects')
plt.ylabel('Average Number of Side Effects')
plt.xlabel('Therapeutic Class')
plt.xticks(rotation=90)
plt.tight_layout()
plt.show()
```

Visualization:

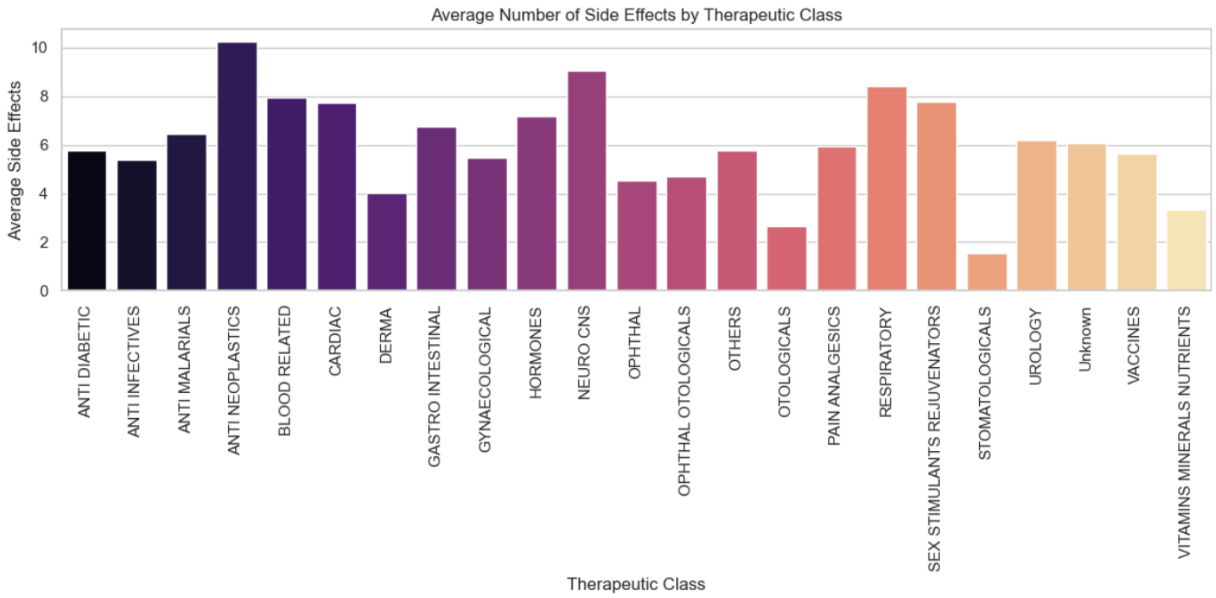


Average Number of Side Effects by Therapeutic Class

Python Code Snippet:

```
# Group by Therapeutic Class and calculate the average side effects
avg_side_effects_by_class = df.groupby('Therapeutic
Class')['side_effect_count'].mean()

# Plot the results
plt.figure(figsize=(12, 6))
sns.barplot(x=avg_side_effects_by_class.index,
y=avg_side_effects_by_class.values, palette="magma")
plt.title('Average Number of Side Effects by Therapeutic Class')
plt.xticks(rotation=90)
plt.ylabel('Average Side Effects')
plt.tight_layout()
plt.show()
```

Visualization:



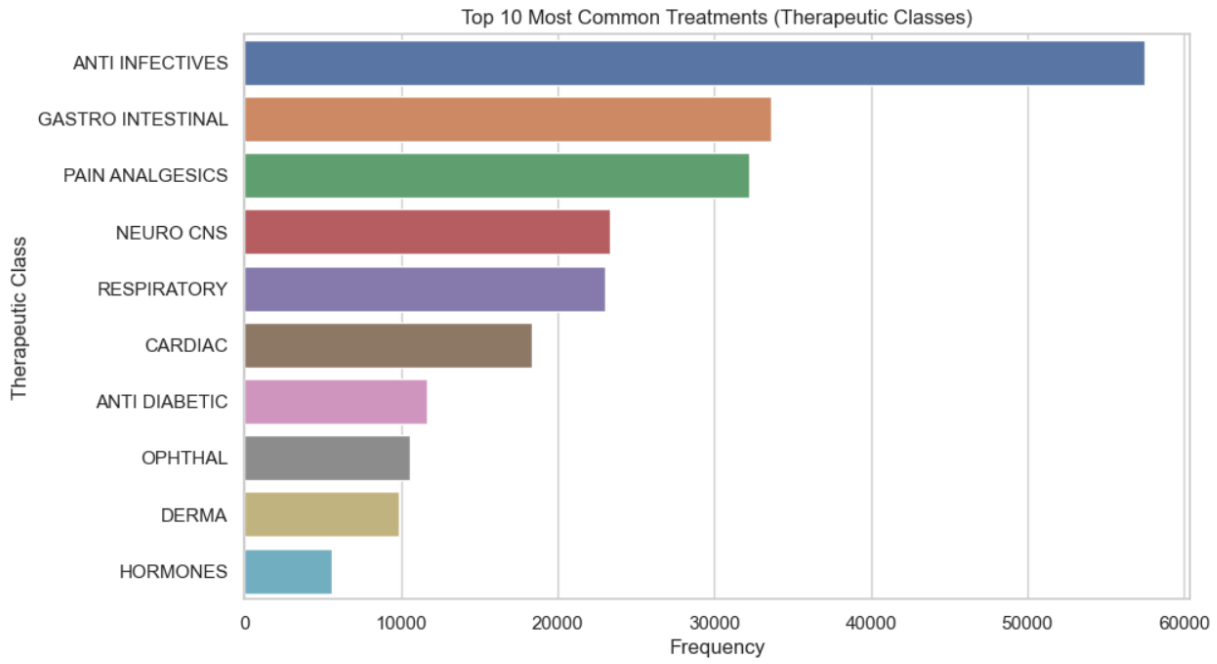
Top 10 Most Common Treatments By Therapeutic Classes

Anti-infectives, Gastro-intestinal, Pain Analgesics are the most common treatments.

Python Code Snippet:

```
# Popular Treatment
# Count the occurrences of each therapeutic class
therapeutic_class_counts = df['Therapeutic Class'].value_counts()

# Plot popular therapeutic classes
plt.figure(figsize=(10,6))
sns.barplot(x=therapeutic_class_counts[:10].values,
y=therapeutic_class_counts[:10].index)
plt.title('Top 10 Most Common Treatments (Therapeutic Classes)')
plt.xlabel('Frequency')
plt.ylabel('Therapeutic Class')
plt.show()
```

Visualization:



Conclusion

Key Insights:

- Neuro CNS and Pain Analgesics therapeutic classes have the highest number of habit-forming drugs.
- Drugs with many substitutes do not necessarily have fewer side effects, indicating that substitutes do not always offer safer alternatives.
- Most drugs in the dataset are not habit-forming, with only 6003 out of 248,218 drugs labeled as habit-forming.

Correct and Precise Interpretation:

The data reveals that certain therapeutic classes, particularly those dealing with neurological and pain management drugs, are more prone to being habit-forming. Additionally, having substitutes does not guarantee that a drug will have fewer side effects, suggesting that further investigation into drug alternatives is required.

Recommendations:

1. Healthcare Providers: Should be cautious when prescribing drugs from habit-forming classes like Neuro CNS and Pain Analgesics.
2. Pharmaceutical Companies: Should focus on developing safer alternatives, particularly for drugs with known side effects and no substitutes.

Future Analysis:

- Side Effect Severity: Further research could explore the severity of side effects across therapeutic classes.
- Dosage Analysis: Investigating how dosage impacts side effects could yield actionable insights.
- Habit-Forming Prediction: Developing models to predict habit-forming tendencies based on chemical and therapeutic classes.