HW 4 Analyzing the Effectiveness of a New Drug Using Statistical Methods

Due Monday, 9/30 on Moodle

Please follow the instructions:

**GITHUB WITH MY COMPLETED CODE:
https://github.com/kerekejo/DS241_Homework4**

1. Fill in blank spots in the code provided in the instructions;
2. Take a screenshot of your completed code by sections with produced results, paste it into this word document;
3. Fill in your discussions when asked.

1. Problem Statement:

- We aim to determine whether the new drug is more effective than a placebo in reducing symptoms of a particular illness.
- We will test the following hypotheses:
    - o **Null Hypothesis (H_0):** There is no significant difference between the effectiveness of the new drug and the placebo.
    - o **Alternative Hypothesis (H_1):** The new drug is significantly more effective than the placebo.

2. Data Collection:

Let's assume you have a CSV file named `drug_trial_data.csv` with the following columns:

- **patient_id**: Unique identifier for each patient.
- **group**: Either "new_drug" or "placebo".
- **pre_treatment_score**: A score measuring the patient's health before treatment (higher is worse).
- **post_treatment_score**: A score measuring the patient's health after treatment (higher is worse).
- **recovery_time**: Number of days taken to recover.
- **side_effects**: Whether the patient experienced side effects (Yes/No).

3. Loading the Data

```
import pandas as pd

# Load the dataset
data = ___________?___________________

# View the first few rows
data.head()
```

Results:

```
import pandas as pd
import matplotlib.pyplot as plt

import seaborn as sns


# Load the dataset

data =
pd.read_csv(r'C:\Users\Joe\OneDrive\Desktop\PythonPrograms\Homework4\simulated_drug_trial_data
.csv')

# View the first few rows

data.head()


#Describe

data.groupby('group').describe()
```

Interpretation: The basic way to open a CSV file in python is to use the pandas library. You must call the read_csv method to open the simulated drug trial data.

4. Exploratory Data Analysis (EDA)

```
# Descriptive statistics for both groups
data.groupby('group').describe()
```

Results:

```
import pandas as pd
import matplotlib.pyplot as plt

import seaborn as sns


# Load the dataset

data =
pd.read_csv(r'C:\Users\Joe\OneDrive\Desktop\PythonPrograms\Homework4\simulated_drug_trial_data
.csv')

# View the first few rows

data.head()


#Describe
```

```
print(data.groupby('group').describe())
```

Console Output:

```
     patient_id                                ... recovery_time
          count   mean      std  min  25%   50%    75% ...       mean     std     min      25%      50%
75%      max
group                                         ...
new_drug     100.0  107.16  62.096328  1.0  54.5  121.0  159.50  ...     6.73567  2.011707  2.056711
5.323368   6.653984   8.080045  13.157762
placebo      100.0   93.84  52.804140  2.0  48.5   93.5  129.25  ...     14.05259  3.168955  5.909340
12.339321  13.914682  16.110789  21.720079


[2 rows x 32 columns]
```

Interpretation: This command helps to allow python to interpret the CSV file that it had previously read in. This method allows for better organization and the usage of data by python from a dataset

Visualizations:

- **Boxplot for Pre-treatment and Post-treatment scores**

```
import seaborn as sns
import matplotlib.pyplot as plt

# Pre-treatment scores
sns.boxplot(x='group', y='pre_treatment_score', data=data)
plt.title('Pre-treatment Scores by Group')
plt.show()

# Post-treatment scores
ns.boxplot(x='group', y='post_treatment_score', data=data )

plt.title("Post-treatment Scores by Group"

plt.show()
```

Results: This box plot visualizes the data gained from the data set. It displays the difference between the pre-treatment health scores (with placebo group included) and the post-treatment health scores (again with placebo group included).
Interpretation: ? Why? One can interpret this box graph as showing that the drug that is being studied is effective when compared to placebo based on pre-treatment and post-treatment scoring.

While the score of the placebo group worsened based on the post-treatment score, the score of the drug group improved after the treatment.

- **Recovery time distribution for both groups**

```
sns.histplot(data=data, x='recovery_time', hue='group', kde=True)
plt.title('Recovery Time Distribution')
plt.show()
```

Results: These lines display a histogram based on the recovery time of the groups. It displays that the group taking the drug had much shorter recovery times as compared to those with placebo.

Interpretation: ? Why? From this graph, one can interpret that the drug is successful in reducing recovery times as compared to the placebo. The graph has two distinct groups, the individuals who have taken the drug, and the group that had taken the placebo. Recovery time is on the X axis, and the count of individuals who recovered is on the Y axis. The graph shows that those who have taken the drug recovered significantly sooner than those who had not.

- **Side effects comparison**

```
sns.countplot(x='side_effects', hue='group', data=data)
plt.title('Side Effects by Group')
plt.show()
```

Results: The result of this data visualization is a bar graph that depicts the side effects of the drug on the two groups, placebo and those who have taken the new drugs. The graph shows that while the majority of people do not face any side-effects, those who have taken the drug did face more than those who did not take the drug.

Interpretation: ? Why? This visualization displays that while the majority of people do not face any side effects, both from the drug and from the placebo, those who do take the drug face more side effects when compared to placebo.

5. Hypothesis Formulation

The hypotheses are:

- **Null Hypothesis (H_0):** The mean recovery time is the same for both groups.
- **Alternative Hypothesis (H_1):** The mean recovery time for the new drug group is significantly different from the placebo group.

6. Statistical Testing

T-Test for Recovery Time:

```
from scipy.stats import ttest_ind

# Split the data into two groups
```

```
new_drug_group = data[data['group'] == 'new_drug']['recovery_time']
placebo_group = data[data[‘group’ == ‘placebo’][‘recovery_time’]

# Perform an independent t-test
t_stat, p_value = ttest_ind(new_drug_group, placebo_group)
print(f"T-statistic: {t_stat}, P-value: {p_value}")
```

Results: T-Statistic: -19.4932619
P-Value: 6.0667X10^-48

Interpretation: ? Why? T-Statistic value allows you to interpret whether or not there is a significant difference between groups. This dataset's T-Statistic value is far from zero and shows that there is a significant difference between the placebo groups and the group that took the new drug. A dataset’s P value is a number that describes how likely it is that data would have occurred under our null hypothesis. This data’s p-value is extremely small, indicating that there is a insignificantly small likelihood that this data would’ve occurred under the null hypothesis.

Chi-Square Test for Side Effects:

A contingency table is a data representation that shows the frequency distribution of two or more categorical variables. It helps to summarize the relationship between these variables by displaying the counts or proportions of observations that fall into each category combination.

```
from scipy.stats import chi2_contingency

# Create a contingency table
contingency_table = pd.crosstab(data['group'], data['side_effects'])
```

The Chi-square test is a statistical method used to determine whether there is a significant association between categorical variables. It compares the observed frequencies in each category of a contingency table to the frequencies we would expect if there were no association between the variables. Overall, the Chi-square test is a powerful tool for exploring relationships in categorical data!

Key Steps in a Chi-square Test:

1) **Formulate Hypotheses**:
   - Null Hypothesis (H0H_0H0): Assumes no association between the variables.
   - Alternative Hypothesis (Ha): Assumes an association exists.
2) **Calculate Expected Frequencies**: Based on the assumption that the null hypothesis is true.
3) **Compute the Chi-square Statistic**:

$$\chi 2 = \sum \frac{(O-E)^2}{E}$$

Where O is the observed frequency and E is the expected frequency.

4) **Determine the Degrees of Freedom**:
    - For the test of independence: (rows−1)×(columns−1).
5) **Compare to the Chi-square Distribution**: Use a Chi-square distribution table to find the p-value or critical value for your test statistic.
6) **Make a Decision**: If the p-value is less than the significance level (usually 0.05), reject the null hypothesis.

```
# Perform the chi-square test
chi2, p, dof, expected = chi2_contingency(contingency_table)
print(f"Chi-Square Statistic: {chi2}, P-value: {p}")
```

- **Interpretation**: If the p-value is less than 0.05, we conclude that there is a significant association between the drug group and the occurrence of side effects.

Results: Chi-Square Statistic: 1.52625, P-Value: 0.21667

Interpretation: ? Why? The hypothesis being tested is based on the side-effects of the two groups based on whether or not they actually took the drug. Based on the chi-square statistic, one can interpret whether or not there is a significant association between the drug group and the occurrence of side effects. This data's P-Value based on the chi-square statistic is 0.21667, and this means that there is not a significant association between the drug group and the occurrence of side-effects.

7. Confidence Interval Calculation

A confidence interval (CI) is a statistical range that estimates where a population parameter, such as a mean or proportion, is likely to fall, based on sample data. It provides a range of values, along with a confidence level that reflects how confident we are that the parameter lies within that range.

Key Components:

1. **Point Estimate**: The sample statistic (e.g., sample mean) that serves as the best estimate of the population parameter.

2. **Margin of Error**: The range around the point estimate that accounts for sampling variability. It's often calculated using a critical value (from the normal or t-distribution) multiplied by the standard error of the estimate.

3. **Confidence Level**: This is usually expressed as a percentage (e.g., 95%, 99%) and indicates the probability that the confidence interval will contain the true population parameter if we were to take many samples.

Formula:

For a mean, a typical confidence interval can be expressed as:

$$CI = Point\ Estimate \pm Margin\ of\ Error$$

Where:

- Margin of Error = $z \times \frac{\sigma}{\sqrt{n}}$ (for a known population standard deviation)

- Or, Margin of Error = $t \times \frac{s}{\sqrt{n}}$ (for an unknown population standard deviation, where s is the sample standard deviation and t is the t-score).

Example:

If a survey finds that the average height of a sample of 100 people is 65 inches with a standard deviation of 4 inches, a 95% confidence interval might be calculated as:

- $CI \;=\; 65 \pm (1.96 \times \frac{4}{\sqrt{100}})$
- CI = 65 ± 0.784
- Resulting in an interval of approximately (64.216,65.784).

```
import numpy as np

# Compute the mean and standard error for both groups
mean_diff = np.mean(new_drug_group) - np.mean(placebo_group)
se_diff = np.sqrt(np.var(new_drug_group)/len(new_drug_group) +
np.var(placebo_group)/len(placebo_group))

# 95% confidence interval
conf_interval = (mean_diff - 1.96 * se_diff, mean_diff + 1.96 * se_diff)
print(f"95% Confidence Interval: {conf_interval}")
```

Results: 95% Confidence Interval: (np.float64(-8.04893105603818), np.float64(-6.584909505937024))

Interpretation: ? Why? This confidence interval indicates that the drug is effective in reducing the recovery time of the patients. The first sign of this is that the confidence interval indicates this is because the interval is negative. This means that the mean of the recovery time for those who take the drug reduces by roughly -8.04 to -6.58. Indicating that our hypothesis is likely correct.

8. Effect Size Measurement (Cohen's d)

```
# Calculate the pooled standard deviation
pooled_std = np.sqrt((np.var(new_drug_group) + np.var(placebo_group)) / 2)

# Calculate Cohen's d
cohen_d = (np.mean(new_drug_group) - np.mean(placebo_group)) / pooled_std
print(f"Cohen's d: {cohen_d}")
```

Results: Cohen's d: -2.770651603468883

Interpretation: ? Why? Cohen's d helps you compare the difference between two groups in terms of standard deviation. From the above result, we can conclude that the placebo and the new drug group's recovery time has a small effect on the hypothesis.

9. Bayesian Probability (Optional) – leave it for future

10. Conclusion and Interpretation

- Based on the **t-test**,
- **Effect Size (Cohen's d)** will tell you the practical significance (larger values indicate stronger effects).
- **Confidence Intervals** provide a range where the true mean difference lies.
- **Chi-square** tells whether side effects are associated with the new drug.

This analysis will give a complete picture of the drug's effectiveness and safety compared to the placebo.

Final words: Based on the T-test, one can determine that there is a significant difference between placebo and drug test groups. We can determine that the probability that this difference would happen under the null hypothesis to be so small it can be seen as insignificant.

Cohen's d allows you to conclude that there is a small effect on the standard deviation when comparing the two hypotheses.

The confidence interval helps you conclude that the mean difference between the placebo and the new drug group is between -8.04 and -6.04. These numbers provide the general range of the difference between the means of the two groups.

Based on the Chi square number and its associated P-value, we can conclude that there is no significant correlation between getting side effects and taking the new drug.

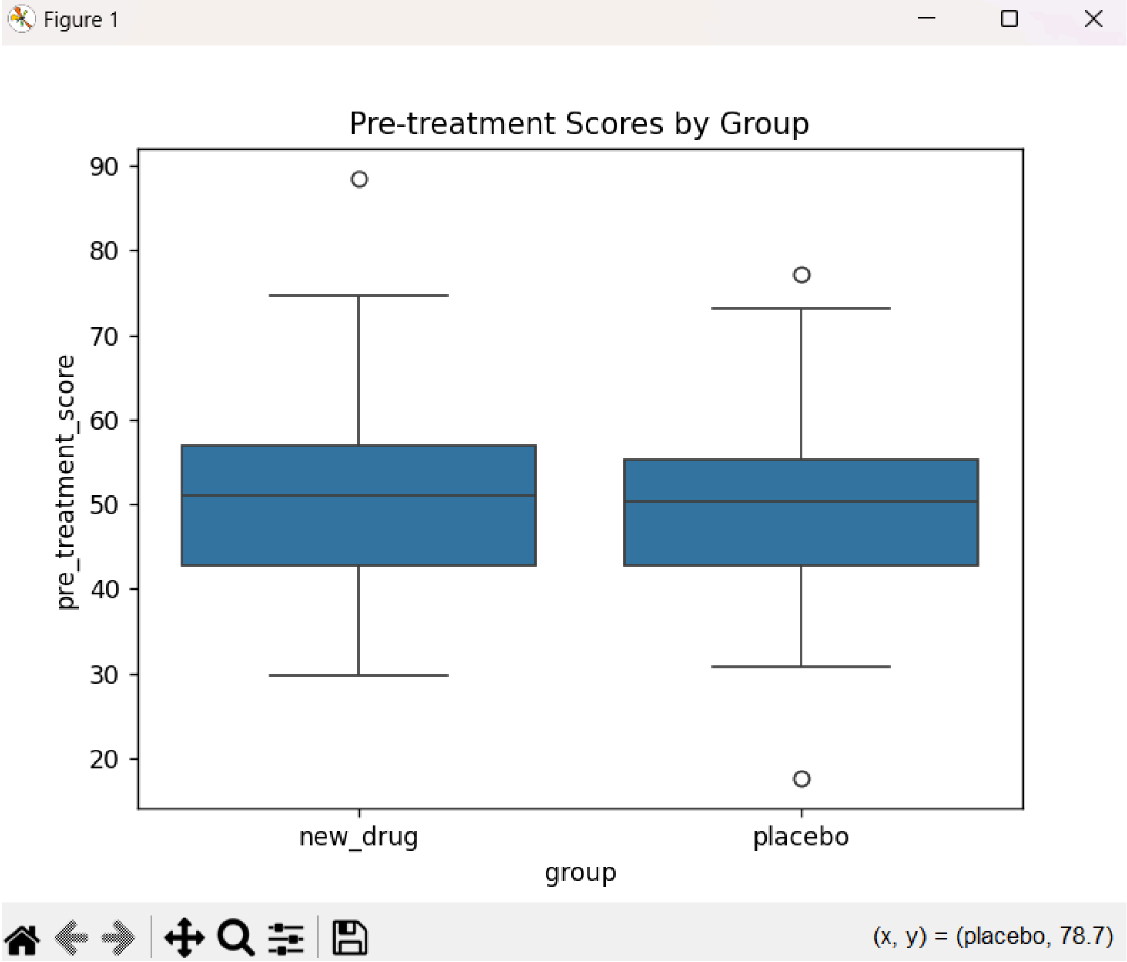


Figure 1
Pre-treatment Scores by Group
pre_treatment_score
90
80
70
60
50
40
30
20
new_drug
placebo
group
(x, y) = (placebo, 78.7)

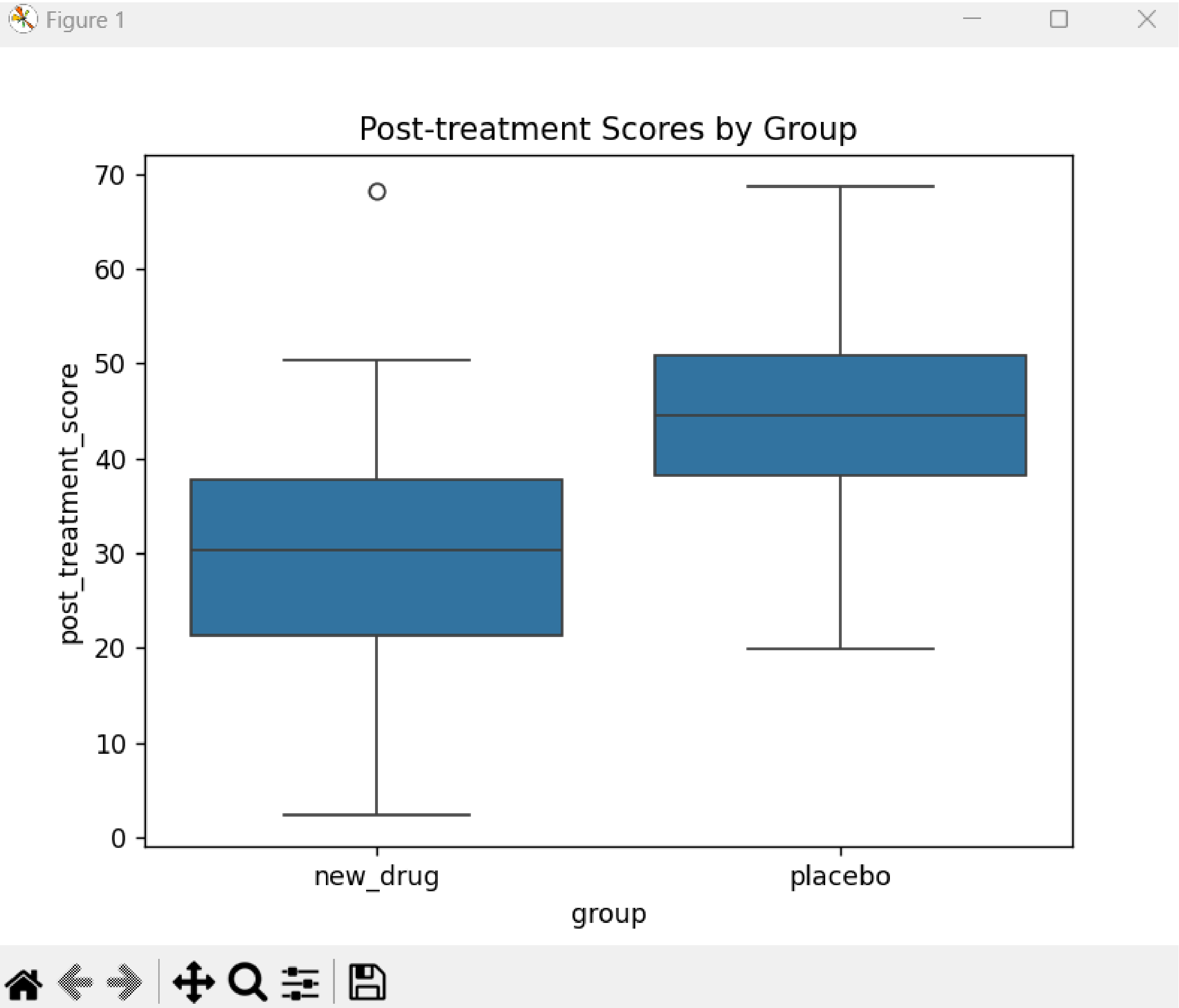


Figure 1
Post-treatment Scores by Group
post_treatment_score
0
10
20
30
40
50
60
70
new_drug
placebo
group

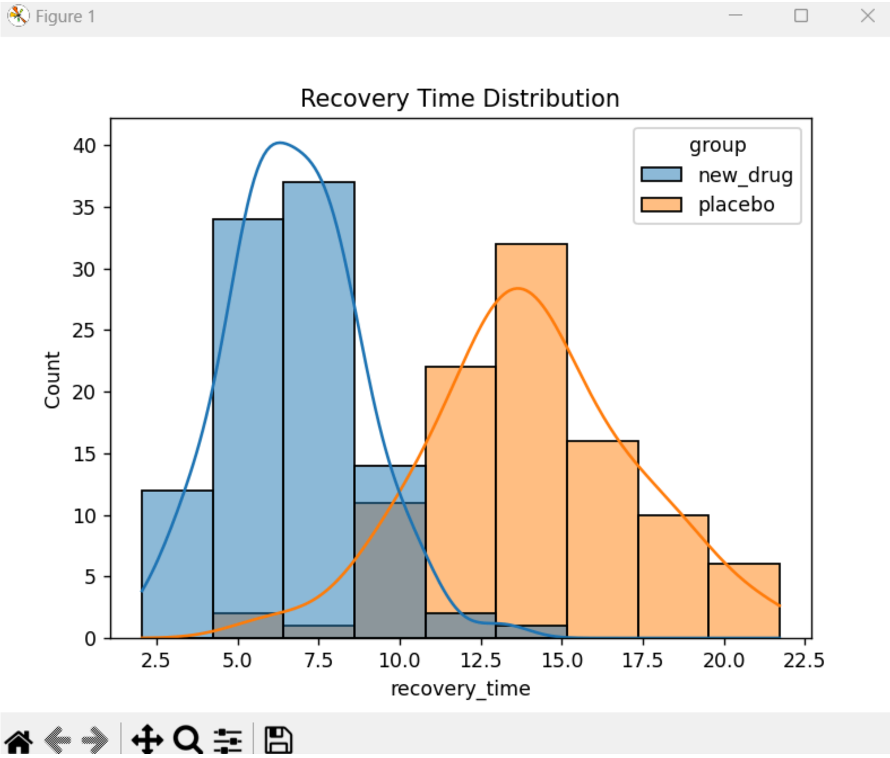


Figure 1
Recovery Time Distribution
group
new_drug
placebo
Count
0
5
10
15
20
25
30
35
40
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
22.5
recovery_time

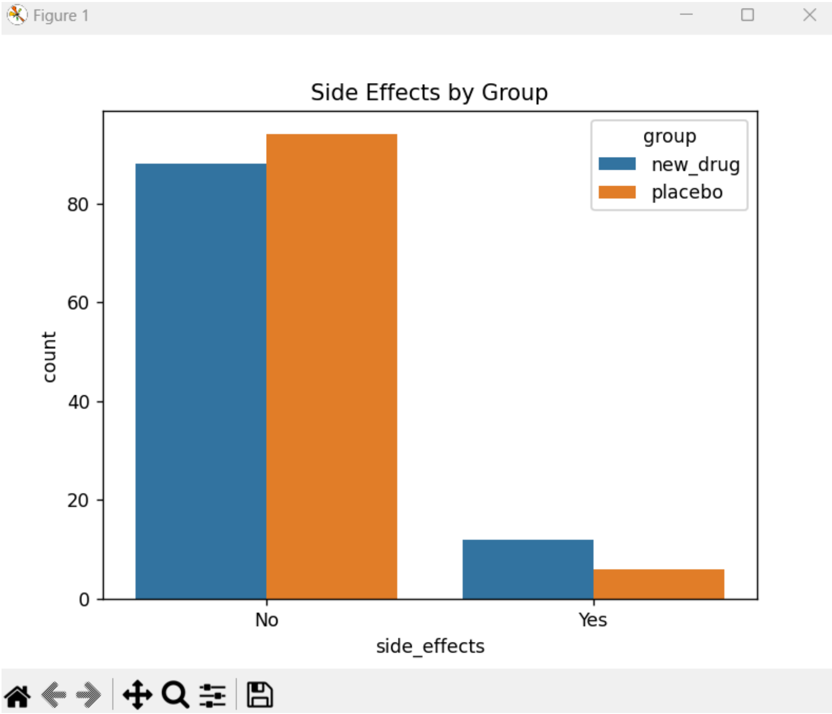
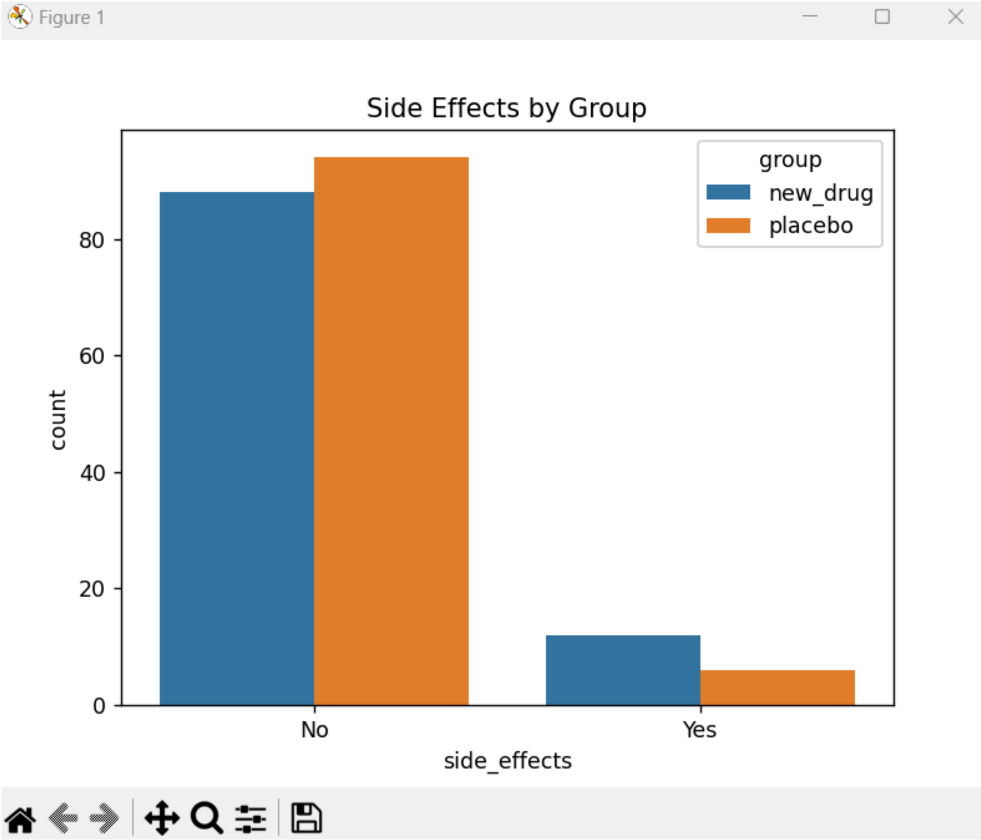
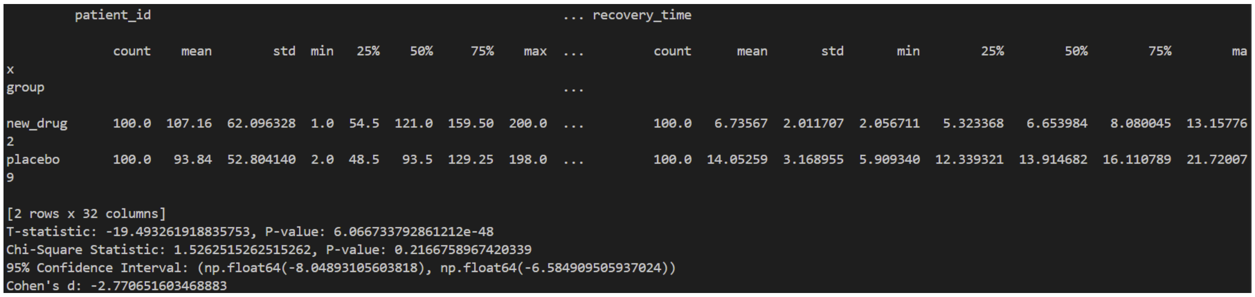


Figure 1
Side Effects by Group
group
new_drug
placebo
count
0
20
40
60
80
No
Yes
side_effects





```
         patient_id                                                      ... recovery_time
              count    mean        std  min   25%    50%     75%    max  ...         count      mean       std       min        25%        50%        75%        ma
x
group                                                                    ...
new_drug      100.0  107.16  62.096328  1.0  54.5  121.0  159.50  200.0  ...         100.0   6.73567  2.011707  2.056711   5.323368   6.653984   8.080045  13.15776
2
placebo       100.0   93.84  52.804140  2.0  48.5   93.5  129.25  198.0  ...         100.0  14.05259  3.168955  5.909340  12.339321  13.914682  16.110789  21.72007
9

[2 rows x 32 columns]
T-statistic: -19.493261918835753, P-value: 6.066733792861212e-48
Chi-Square Statistic: 1.5262515262515262, P-value: 0.2166758967420339
95% Confidence Interval: (np.float64(-8.04893105603818), np.float64(-6.584909505937024))
Cohen's d: -2.770651603468883
```