Advanced Mathematical Statistics (MTH 522)

Project 1

Investigating Statistical Significance and Effect Size of Crab Shells Molting Data

Submitted By:

Supreeth Mohan - 02036295

Roshni Pal - 02137180

Trina Xavier - 02102403

Aryan Bhalla – 02107402

The Issues:

The crab molt data consists of " postmolt " and "premolt" sizes of the (shells) of 473 Dungeness crabs. In our analysis the primary issues we are delving into are:

• Is there a statistically significant difference in size of crabs before and after molting? As the variability in growth patterns could affect the observed changes in shell size before and after molting, potentially influencing the interpretation of results.

• Investigating whether age of the crabs influences the size difference between premolt and post molt shells could help understand the developmental dynamics of crabs.

• Statistical analyses of the dataset reveals discernible trends indicating potential changes in crab shell size before and after molting.

Findings:

Based on the analysis of the crab data, key components of this dataset include measurements of shell sizes premolt and post-molt, providing essential insights into the molting dynamics of Dungeness crabs. the findings include:

In comparing the premolt and postmolt data of Dungeness crabs, a **Cohen's d value** of -0.96206 was calculated, indicating a large effect size. This suggests a significant difference in shell sizes between premolt and postmolt crabs, Crabs undergo noticeable changes in shell dimensions following molting, with postmolt shells typically averaging around 15mm, highlighting a critical aspect of crab development

The Monte Carlo method for crab molt data will simulate multiple random samples based on the observed data distribution, allowing for the assessment of statistical uncertainty and the estimation of parameters for hypothesis testing outcomes.

Discussion:

The moderate effect size observed, coupled with the statistical significance of the results, underscores the biological significance of the molting process in crab growth. The consistent increase in size across the population suggests that molting plays a pivotal role as a growth strategy for crabs, likely contributing to their survival, competitive advantage, and reproductive success.

Other demographics such as climate change, habitat, etc exert influence on the extent of growth during molting. Moreover, the findings hold implications for understanding crab population

dynamics, fisheries management, and the formulation of effective conservation strategies, urging further exploration in these domains.

Appendix A: METHOD

1. **Data collection :** The data used in this study was obtained for both after and before molting provided below.
   https://www.dropbox.com/scl/fi/ezylkm0vw29y7jufn3y9w/Crab_molt.csv?rlkey=4gfgcgvsos2z0ipz1gzj4ax6d&dl=0

2. **Data Preparation:** Crab Molting data were downloaded and examined, and the procedure was documented.

3. **Variable Creation:** The two variables in the analysis are "post_molt" and "pre_molt" which represents the size of the crabs before and after molting, respectively.

Analytic Method:

The statistical procedures used in the above analysis are as follows:

• **Descriptive statistics:** Smooth histograms of crab molt data representing the distribution of shell sizes before and after molting, offering insight into the central tendency and variability of these measurements across the population of Dungeness crabs

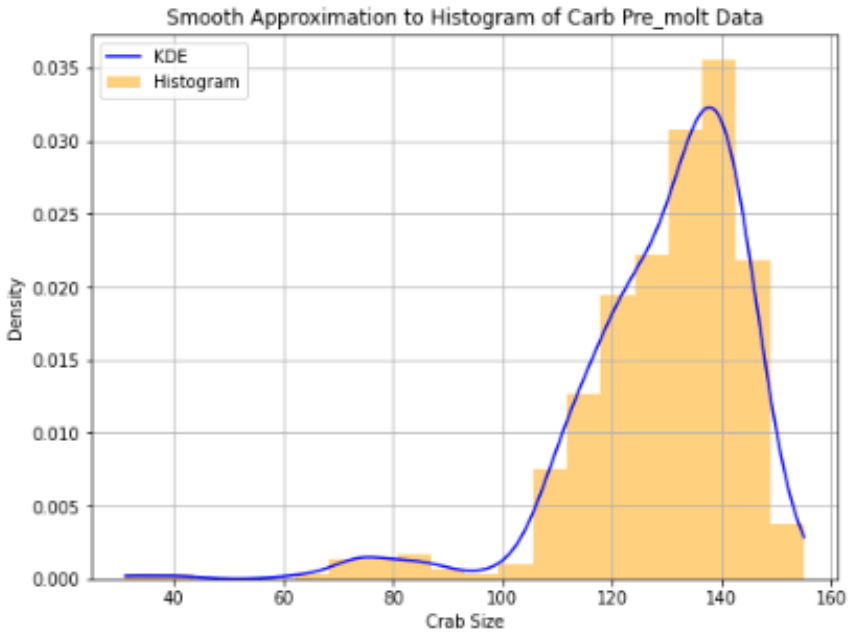
• **CDF**:This graphical representation aids in assessing the overall distribution and variability of shell sizes within the population of Dungeness crabs.

• **Cohen's d:** The calculated Cohen's d value quantifies the effect size of the difference in shell sizes between pre_molt and post_molt crabs.

• **Monte Carlo Method :**The Monte Carlo method for crab molt data will simulate multiple random samples based on the observed data distribution, allowing for the assessm hypothesis testing outcomes.

APPENDIX B: RESULT

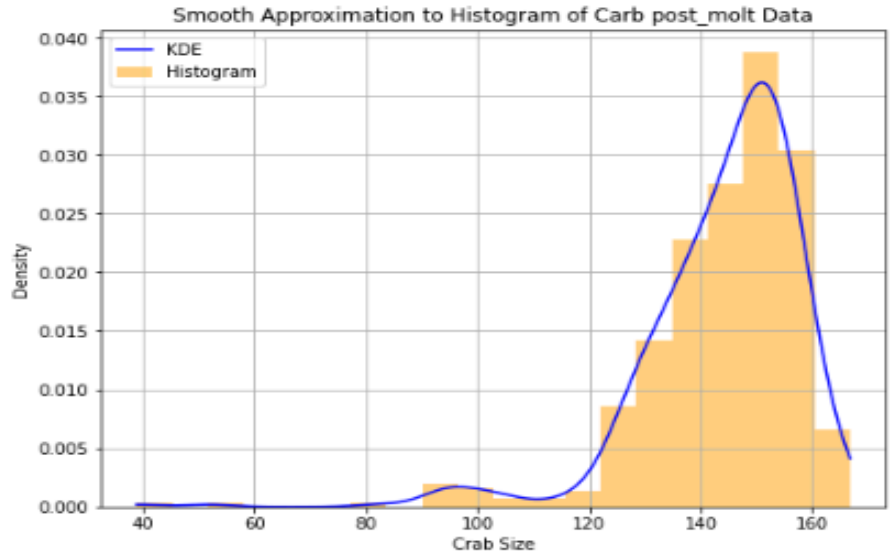
- **Smooth Approximation of Histogram of Crab Pre_molt Data:**




```
Minimum of pre_molt = 31.1
Maximum of pre_molt = 155.1
Median of pre_molt = 132.8
Mean of pre_molt = 129.21186440677965
Skewness of pre_molt = -2.0098801542639126
Kurtosis of pre_molt = 6.851369760728378
```

Fig 1.

- **Smooth Approximation of Histogram of Crab Post_molt Data:**




```
Minimum of post_molt = 38.8
Maximum of post_molt = 166.8
Median of post_molt = 147.4
Mean of post_molt = 143.89766949152542
Skewness of post_molt = -2.354390947858422
Kurtosis of post_molt = 10.23684737910977
```

Fig 2

- **Comparison of Distributions of Crab_Pre_molt and Post_molt:**

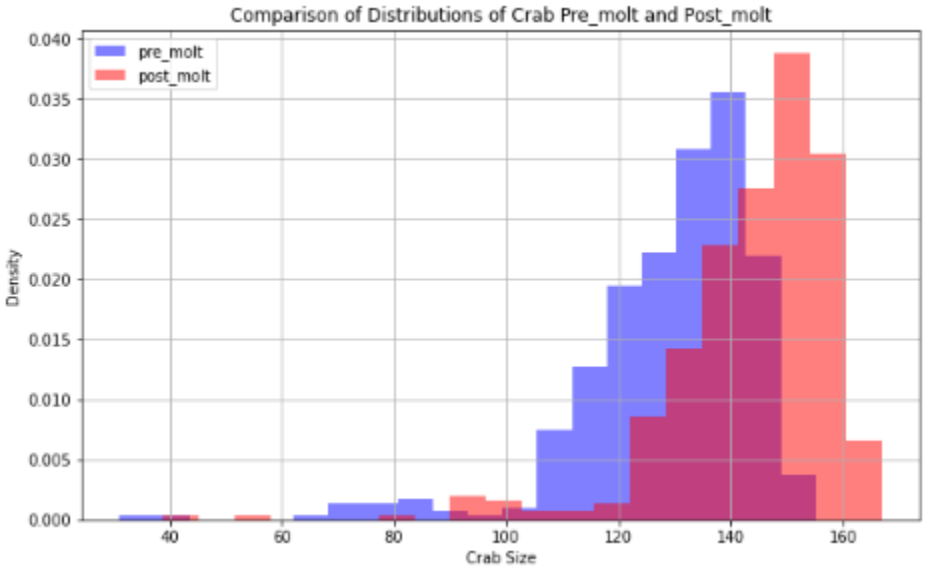
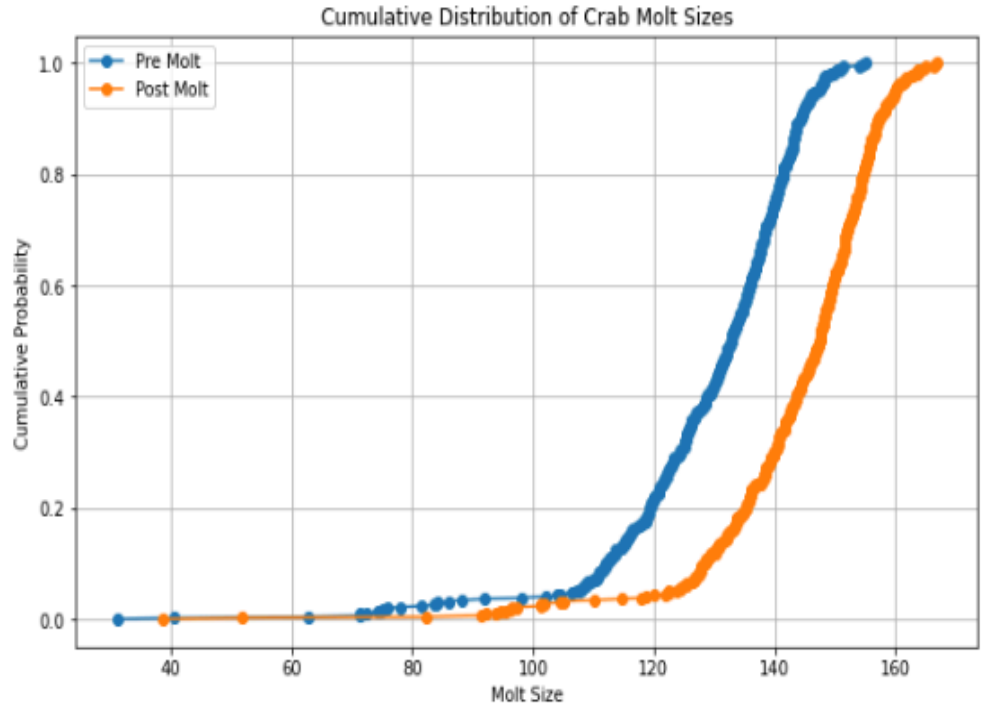


Fig 3

- **Cumulative Distribution of Crab Molt Data:**




```
Mean pre-molt size: 129.21186440677965
Mean post-molt size: 143.89766949152542
Effect size (Difference in mean sizes): -14.685805084745766
```

Fig 4

The graph shows the cumulative distribution of crab molt sizes. The x-axis shows the molt size, while the y-axis shows the cumulative probability. The blue line represents the pre-molt size, and the red line represents the post-molt size.

Here are some key observations about the image:

- Most crabs increase in size after molting. This is shown by the fact that the red line (post-molt) is generally higher than the blue line (pre-molt) for most molt sizes.
- There is a wider range of sizes after molting. The red line is steeper than the blue line, indicating that there is more variability in post-molt sizes than pre_molt sizes.
- There is a limit to how much crabs can grow. The lines flatten out at larger molt sizes, suggesting that there is a maximum size that crabs can reach.

- **Cohen's D:**

```
Cohen's d: -0.9620678692342478
```

Fig 5

Cohen's d is -0.96206, we can interpret that the mean size of post_molt crab is higher than the mean size of pre_molt crab.

- **Q-Q Plot of Crab for Pre_molt Data:**

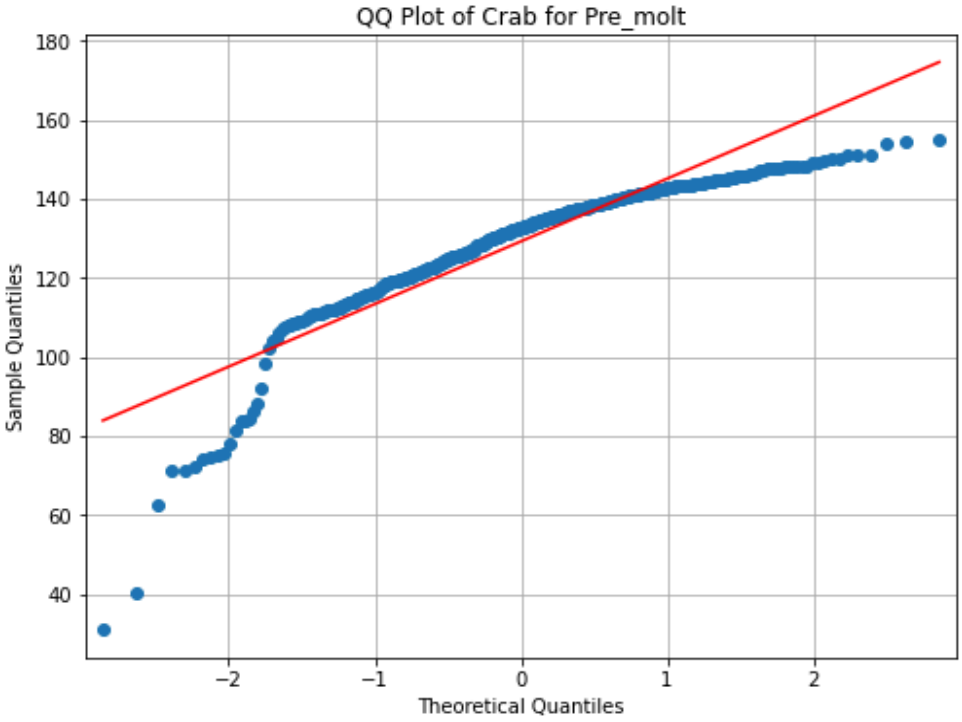


Fig 6

The x-axis shows the sample quantiles, while the y-axis shows the theoretical quantiles. The points do not fall on a straight line, which suggests that the two distributions are not the same.

Specifically, the fact that the points curve upwards in the center of the plot suggests that there are more crabs than expected with quantiles in the middle of the range.

- **Q-Q Plot for Crab Post_molt Data:**

Similar to pre_molt data, we have plotted the Q-Q plot for post_molt data as well.

- **Monte Carlo:**

```
Observed Mean Crab Size Difference: -14.685805084745766
P-value (Monte Carlo): 0.0
Time taken to estimate the p-value using Monte-Carlo method is 49.41552 seconds
```

Fig 7

It shows the results of a study on the difference in mean crab size between two groups. The p-value is a statistical measure that indicates the probability of observing the data, or more extreme data. The p-value is 0.0, which means that there is very strong evidence to reject the null hypothesis and conclude that there is a statistically significant difference in mean crab size between the two groups. It also shows that the time taken to estimate the p-value using the Monte Carlo method was 49.41552 seconds

Appendix C: DATA AND CODE

Smooth Approximation of Crab Pre_molt Size Distribution

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
from scipy.stats import gaussian_kde

# Load the data from the CSV file
data = pd.read_csv('Crab_molt.csv')

# Filter data for pre_molt
df = pd.DataFrame(data)
df_cleaned = df.dropna()

post_molt = df_cleaned['post molt size'].sort_values().to_numpy()

# Set up the figure and axis
plt.figure(figsize=(8, 6))

# Create a kernel density estimation
kde = gaussian_kde(post_molt)

# Create a range of x values for the plot
x_values = np.linspace(min(post_molt), max(post_molt), 1000)

# Plot the kernel density estimate
plt.plot(x_values, kde(x_values), color='blue', label='KDE')

# Plot the histogram for comparison
plt.hist(post_molt, bins=20, density=True, color='orange', alpha=0.5, label='Histogram')

# Add labels and legend
plt.title('Smooth Approximation to Histogram of Carb post_molt Data')
plt.xlabel('Crab Size')
plt.ylabel('Density')
plt.legend()

# Show the plot
plt.grid(True)
plt.show()


# Calculate statistical information for the 'pre_molt' column
post_molt_stats = df['post molt size'].describe()
post_molt_skewness = df['post molt size'].skew()
post_molt_kurtosis = df['post molt size'].kurtosis()

# Print the statistical information
print("Minimum of post_molt =", post_molt_stats['min'])
print("Maximum of post_molt =", post_molt_stats['max'])
print("Median of post_molt =", post_molt_stats['50%'])
print("Mean of post_molt =", post_molt_stats['mean'])
print("Skewness of post_molt =", post_molt_skewness)
print("Kurtosis of post_molt =", post_molt_kurtosis)

# Show the plot
plt.show()
```

Smooth Approximation of Crab Post_molt Size Distribution

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
from scipy.stats import gaussian_kde

# Load the data from the CSV file
data = pd.read_csv('Crab_molt.csv')

# Filter data for pre_molt
df = pd.DataFrame(data)
df_cleaned = df.dropna()

pre_molt = df_cleaned['pre molt size'].sort_values().to_numpy()

# Set up the figure and axis
plt.figure(figsize=(8, 6))

# Create a kernel density estimation
kde = gaussian_kde(pre_molt)

# Create a range of x values for the plot
x_values = np.linspace(min(pre_molt), max(pre_molt), 1000)

# Plot the kernel density estimate
plt.plot(x_values, kde(x_values), color='blue', label='KDE')

# Plot the histogram for comparison
plt.hist(pre_molt, bins=20, density=True, color='orange', alpha=0.5, label='Histogram')

# Add labels and legend
plt.title('Smooth Approximation to Histogram of Carb Pre_molt Data')
plt.xlabel('Crab Size')
plt.ylabel('Density')
plt.legend()

# Show the plot
plt.grid(True)
plt.show()


# Calculate statistical information for the 'pre_molt' column
pre_molt_stats = df['pre molt size'].describe()
pre_molt_skewness = df['pre molt size'].skew()
pre_molt_kurtosis = df['pre molt size'].kurtosis()

# Print the statistical information
print("Minimum of pre_molt =", pre_molt_stats['min'])
print("Maximum of pre_molt =", pre_molt_stats['max'])
print("Median of pre_molt =", pre_molt_stats['50%'])
print("Mean of pre_molt =", pre_molt_stats['mean'])
print("Skewness of pre_molt =", pre_molt_skewness)
print("Kurtosis of pre_molt =", pre_molt_kurtosis)

# Show the plot
plt.show()
```

Comparison of Distribution of Crab Pre_molt and Post_molt

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


# Load the data from the CSV file
data = pd.read_csv('Crab_molt.csv')

# Filter data for pre_molt
df = pd.DataFrame(data)
df_cleaned = df.dropna()

pre_molt = df_cleaned['pre molt size'].sort_values().to_numpy()
post_molt = df_cleaned['post molt size'].sort_values().to_numpy()

# Set up the figure and axis
plt.figure(figsize=(10, 6))

# Create histogram for black ages
plt.hist(pre_molt, bins=20, alpha=0.5, color='blue', label='pre_molt', density=True)

# Create histogram for white ages
plt.hist(post_molt, bins=20, alpha=0.5, color='red', label='post_molt', density=True)

# Add labels and legend
plt.title('Comparison of Distributions of Crab Pre_molt and Post_molt')
plt.xlabel('Crab Size')
plt.ylabel('Density')
plt.legend()

# Show the plot
plt.grid(True)
plt.show()
```

Cumulative Distribution function(CDF):

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

# Load data from CSV file
data = pd.read_csv('Crab_molt.csv')

df = pd.DataFrame(data)
df_cleaned = df.dropna()

# Separate data for pre_molt and post_molt
pre_molt = df_cleaned['pre molt size'].sort_values().to_numpy()
post_molt = df_cleaned['post molt size'].sort_values().to_numpy()

# Calculate cumulative distribution functions (CDFs)
pre_molt_cdf = np.arange(len(pre_molt)) / (len(pre_molt) - 1)
post_molt_cdf = np.arange(len(post_molt)) / (len(post_molt) - 1)

# Plotting
plt.figure(figsize=(10, 6))

plt.plot(pre_molt, pre_molt_cdf, label='Pre Molt', marker='o')
plt.plot(post_molt, post_molt_cdf, label='Post Molt', marker='o')

plt.title('Cumulative Distribution of Crab Molt Sizes')
plt.xlabel('Molt Size')
plt.ylabel('Cumulative Probability')
plt.grid(True)
plt.legend()

plt.show()


# Effect size calculation
mean_pre_molt = np.mean(pre_molt)
mean_post_molt = np.mean(post_molt)
effect_size = mean_pre_molt - mean_post_molt

print("Mean pre-molt size:", mean_pre_molt)
print("Mean post-molt size:", mean_post_molt)
print("Effect size (Difference in mean sizes):", effect_size)
```

Cohen's D:

```
import pandas as pd
import numpy as np

# Load data from CSV file
data = pd.read_csv('Crab_molt.csv')

df = pd.DataFrame(data)
df_cleaned = df.dropna()

# Separate data for pre_molt and post_molt
pre_molt = df_cleaned['pre molt size'].sort_values().to_numpy()
post_molt = df_cleaned['post molt size'].sort_values().to_numpy()

# Calculate sample sizes
n1 = len(pre_molt)
n2 = len(post_molt)

# Calculate sample variances
s1_sq = np.var(pre_molt, ddof=1)
s2_sq = np.var(post_molt, ddof=1)

# Calculate pooled standard deviation
s_pooled = np.sqrt(((n1 - 1) * s1_sq + (n2 - 1) * s2_sq) / (n1 + n2 - 2))

# Calculate Cohen's d
cohen_d = (np.mean(pre_molt) - np.mean(post_molt)) / s_pooled

print("Cohen's d:", cohen_d)
```

Q-Q Plot for Crab Pre_molt data:

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import statsmodels.api as sm

# Load the data from the CSV file
data = pd.read_csv('Crab_molt.csv')

# Filter data for pre_molt
df = pd.DataFrame(data)
df_cleaned = df.dropna()

pre_molt = df_cleaned['pre molt size'].sort_values().to_numpy()

# Plotting
plt.figure(figsize=(8, 6))

fig, ax = plt.subplots(figsize=(8, 6))
sm.qqplot(pre_molt, line='s', ax=ax, color='skyblue')

plt.title('QQ Plot of Crab for Pre_molt')
plt.xlabel('Theoretical Quantiles')
plt.ylabel('Sample Quantiles')
plt.grid(True)
plt.show()
```

Q-Q Plot for Crab Post_molt data:

Similar to pre_molt data, we have written the code for the Q-Q plot for post_molt data as well.

Monte-Carlo Method For Estimating The P-Value:

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

# Load data from CSV file
data = pd.read_csv('Crab_molt.csv')

df = pd.DataFrame(data)
df_cleaned = df.dropna()

# Separate data for pre_molt and post_molt
pre_molt = df_cleaned['pre molt size'].sort_values().to_numpy()
post_molt = df_cleaned['post molt size'].sort_values().to_numpy()

# Calculate cumulative distribution functions (CDFs)
pre_molt_cdf = np.arange(len(pre_molt)) / (len(pre_molt) - 1)
post_molt_cdf = np.arange(len(post_molt)) / (len(post_molt) - 1)

# Plotting
plt.figure(figsize=(10, 6))

plt.plot(pre_molt, pre_molt_cdf, label='Pre Molt', marker='o')
plt.plot(post_molt, post_molt_cdf, label='Post Molt', marker='o')

plt.title('Cumulative Distribution of Crab Molt Sizes')
plt.xlabel('Molt Size')
plt.ylabel('Cumulative Probability')
plt.grid(True)
plt.legend()

plt.show()


# Effect size calculation
mean_pre_molt = np.mean(pre_molt)
mean_post_molt = np.mean(post_molt)
effect_size = mean_pre_molt - mean_post_molt

print("Mean pre-molt size:", mean_pre_molt)
print("Mean post-molt size:", mean_post_molt)
print("Effect size (Difference in mean sizes):", effect_size)
```

Reference

[1] *MTH 522 (Advanced Mathematical Statistics, sections 02)*

https://mth522.wordpress.com/

[2]https://www.dropbox.com/scl/fi/ezylkm0vw29y7jufn3y9w/Crab_molt.csv?rlkey=4gfgcgvsos2z0ipz1gzj4ax6d&dl=0

Contributions :

Together, the four of us contributed equally and conducted a comprehensive study of the dataset.

Supreeth Mohan: Worked on the Issues, Discussion, Methods, Data Cleaning, Code, and Results sections. Also, used graphs to analyze the data using the various methods discussed in the report.

Trina Xavier: Worked on identifying issues, writing code for the analysis models, and producing the graphs.

Aryan Bhalla: Worked on the Issues, Findings, and Result sections. Plotted graphs and used various models and tests to analyze the data as discussed.

Roshni Pal: Worked on initial analyses, analyzing and looking for different models to fit the data on, testing various fits for their errors.