UMass | Dartmouth

Advanced Mathematical Statistics

Project 1

"Examining Police Shootings: A Comprehensive Analysis On Age and Race Dynamics among the Victims of United States"

Submitted By:

Supreeth Mohan - 02036259
Roshni Pal - 02137180
Trina Xavier - 02102403
Aryan Bhalla – 02107402

The Issues:

In this project, we delved deeper into analyzing and statistically interpreting the police shooting data. The data consists of the ages of black people, and of white, the people shot and killed by police in the United States. Our objective for this project is to uncover meaningful statistical insights. Below are the primary issues we are examining:

1. Examining patterns in the mean age of individuals involved in police shootings and exploring its correlation with race for understanding the factors contributing to these similarities.
2. The data can provide an understanding of age-related disparities in police shootings to understand potential age-related biases in law enforcement interactions and inform equitable policing reforms.
3. The absence of reporting on crimes, including incidents of shootings, has a significant impact on the accuracy and comprehensiveness of available data, and in what ways does this absence influence our understanding of the true extent and nature of these incidents?
4. In the given dataset which race appears to be more frequently victimized?

Findings:

1. **Age Distribution** – By comparing the cumulative distribution of ages, we got to know that the mean age of black people being shot by police is 33.0486 ~ 33, and the mean age of white people being shot by police is 40.278 ~ 40. The estimated Effect size between the mean age for black people shot by police and white people shot by police is -7.222. A larger effect size (either positive or negative), **Negative,** in this context, suggests a more substantial difference in age between the two races.
2. **Race Distribution** - A positive Cohen's d indicates that the mean age of black people shot by police is higher than the mean age of white people shot by police, while a negative Cohen's d indicates the opposite. As we have found that, ***Cohen's d*** is **-0.57542**, we can interpret that the mean age of white people shot by police is higher than the mean age of black people shot by police. In other words, there is a considerable difference in age, with black individuals being considerably younger than white individuals when involved in police shootings.
3. **Mean Age -** In this context, the Monte Carlo method is used to estimate statistical significance when comparing the mean age difference between black and white individuals. Below is the explanation of the findings:

Mean age difference for Black and White people = -7.222270824175077: This indicates that, on average, the age of black people shot by police is approximately 7.22 years lower than the age of white people shot by police.

Number of Samples with mean difference > -7.22227: 1999999: Out of the 2,000,000 Monte Carlo simulations conducted, all of them resulted in a mean difference between black and white individuals that was greater than the observed mean difference of -7.22227 years.

P-value: 0.0: The p-value is the proportion of permutations where the difference in mean ages is greater than or equal to the observed difference. This means that in all the permutations conducted, the observed mean difference of -7.22227 years was never exceeded. A p-value of 0.0 suggests that there is no statistical significance in the observed difference in mean ages between black and white individuals shot by the police.

Discussion:

The analysis of police shooting data has revealed interesting insights with implications for public awareness strategies and interventions:

Comprehensive Data Patterns Analysis: Race Analysis, and Age Distribution, examining the utilization of multiple analytical approaches for a holistic understanding.

Effect Size:
We analyzed the dataset, evaluating the impact size through two approaches: (i) a comparison of cumulative age distributions and (ii) Cohen's d. Our findings indicate a medium effect size in the estimated difference between the mean age of black individuals shot by police and the mean age of white individuals shot by police.

Statistical Practices in Age Distribution:
Monte Carlo Method, Demonstrating a commitment to robust statistical practices in age distribution analysis. Acknowledging and addressing deviations from normality to ensure the reliability of conclusions drawn from age-related data patterns.

Overall, these analyses contribute to the ongoing dialogue on police shootings by uncovering patterns, discrepancies, and potential areas for reform. They provide a foundation for evidence-based discussions surrounding law enforcement practices, bias mitigation, and engagement.

Appendix A: METHOD

We sourced the fatal police shootings dataset from the provided class link, importing it into a Jupyter Notebook. In our analysis, we amalgamated the 'age' and 'race' columns, leading to an exploration of the intricate relationship between them.

1. **Data Collection:**
   The data used in this study was obtained from The Police Shooting Dataset provided below.
   https://www.dropbox.com/scl/fi/3ukmq88kh4uowplv8f6z7/Police-Shootings-Age-Race.csv?rlkey=cic8y3kcrgf0lq18jujsuek2d&dl=0

2. **Data Preparation:**
   Police shooting data were downloaded and examined, procedure was documented.

3. **Variable Creation:**
   Using Pandas, we retrieved the .csv file– Police Shootings Age Race.csv and loaded it to the variable called data.
   In that .csv file, the data set has age and race, columns then from the race column data for black people is extracted as black_ages, and the same goes for white as white_ages.

Analytic Method:

1. **Descriptive statistics** provides a summary of key features in a dataset, while a histogram visually represents the distribution of data, offering insights of shape and tendencies.

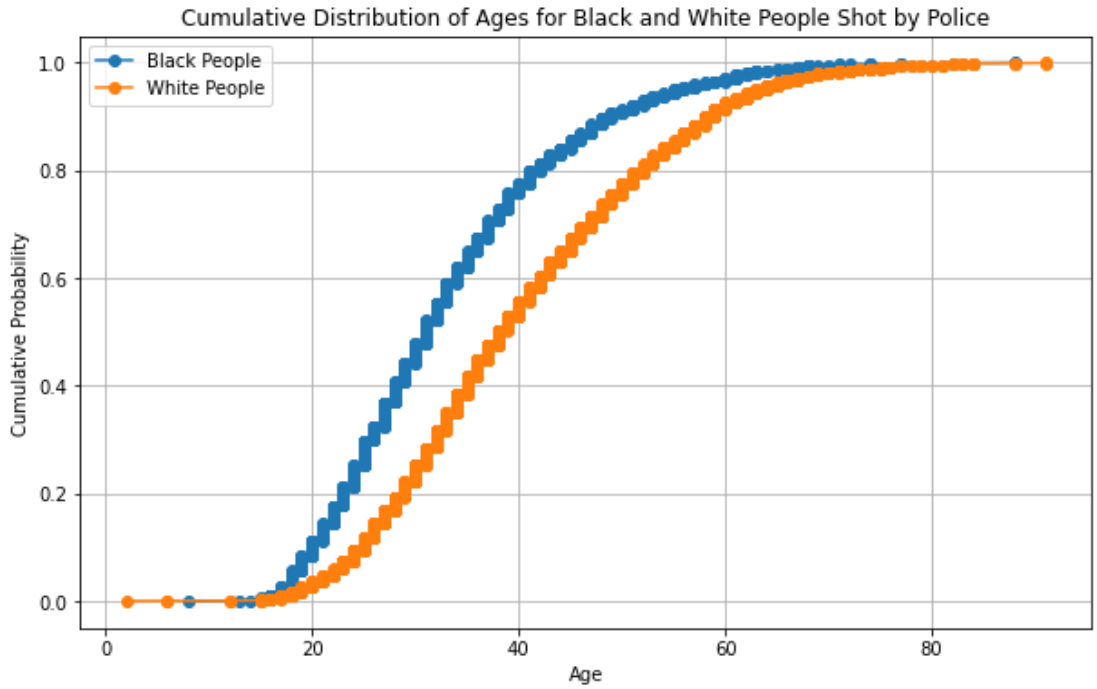
2. **Cohen's d** - It is a statistical measure used to quantify the effect size of the difference between two groups in a study. It's particularly useful in experimental and observational studies where researchers want to understand the magnitude of differences between groups beyond the statistical significance.

3. **Monte Carlo Method** - Due to deviations from normality, the Age distribution of individuals killed by police and the correctness of the computed p-value, will utilize the Monte Carlo method to get the p-value. Monte Carlo methods employ random sampling to solve complex problems. They simulate system behavior by sampling from probability distributions representing parameter uncertainties. Through repeated sampling, Monte Carlo methods estimate integrals, probabilities, and expected values by averaging results. As sample size increases, estimate accuracy improves.

4. **CDF** - Using the Cumulative distribution function (CDF) shows the distribution of age and race, to describe the probability distribution of random variables in a table.

APPENDIX B: RESULT

- **Cumulative Distribution of Ages for Black and White People**




```
Mean age for black people shot by police: 33.04860442733398
Mean age for white people shot by police: 40.270875251509054
Effect size (Difference in mean ages): -7.222270824175077
```

The blue line represents the mean age for Black people shot by police, while the orange line represents the mean age for White people shot by police. The graph shows that Black people shot by police tend to be younger than White people shot by police. The mean age for Black people shot by police is 33.05, while the mean age for White people shot by police is 40.27.

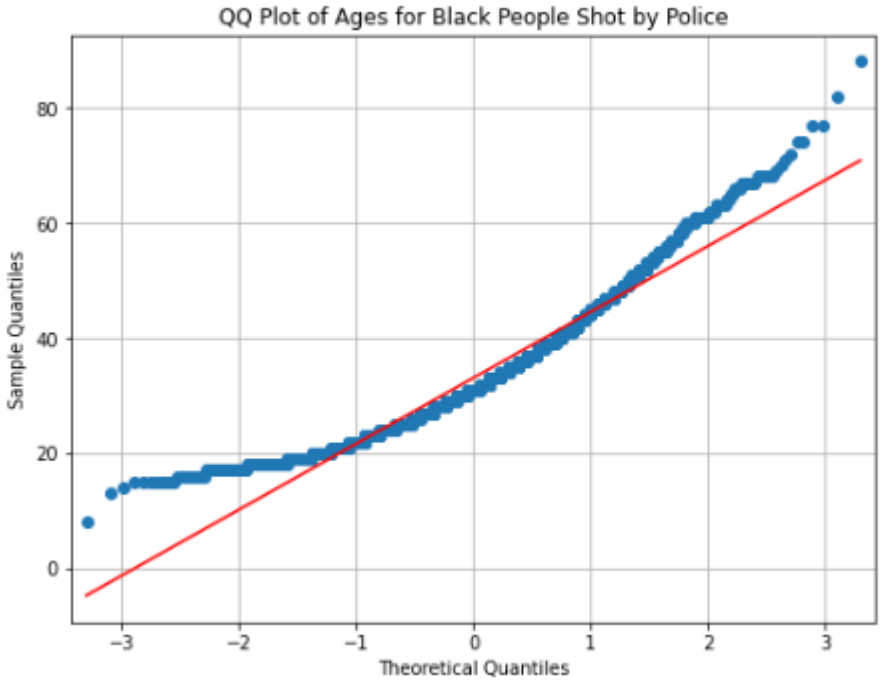
The effect size, which is the difference in mean ages, is -7.22. This means that Black people shot by police are, on average, 7.22 years younger than White people shot by police.

- **Cohen's D:**

```
Cohen's d: -0.575422332150653
```

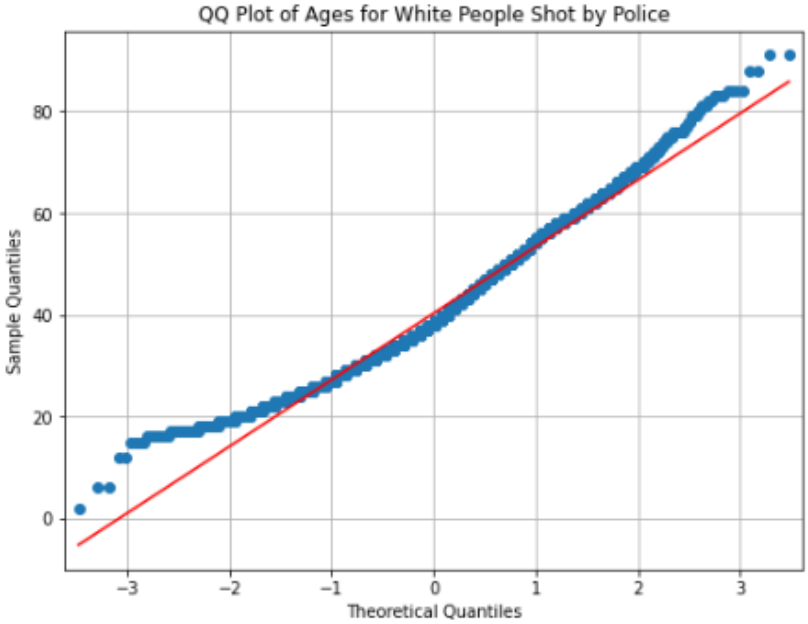
Cohen's d is -**0.57542**, we can interpret that the mean age of white people shot by police is higher than the mean age of black people shot by police.

Q-Q Plot of Ages for Black People shot by police in the United States



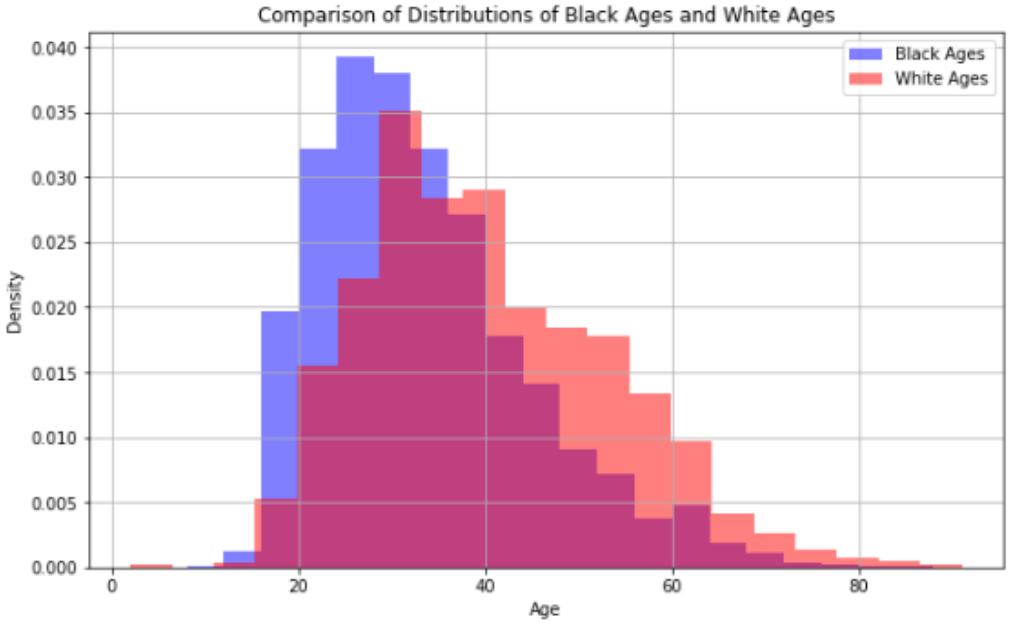
The red line in the plot represents the expected distribution of ages, based on the US population data. The blue line represents the actual distribution of ages of Black people shot by police. The plot shows that there is a significant difference between the two distributions.

Q-Q Plot of Ages for White People shot by police in the United States



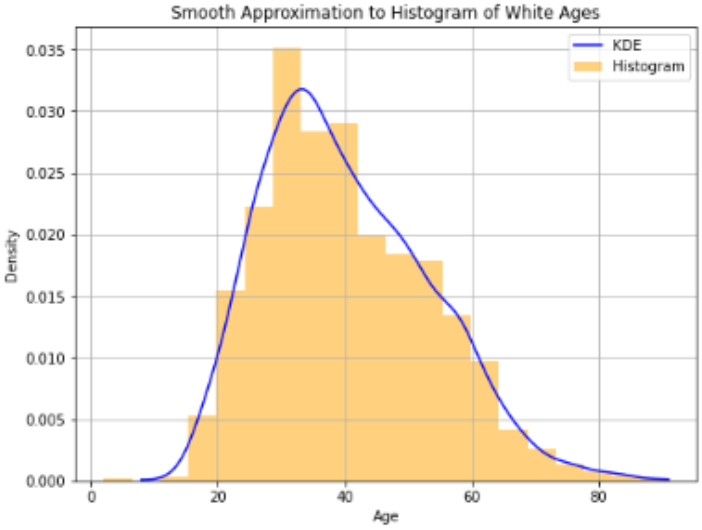
Comparison of Distributions of Ages

This graph allows for a comparison between the age groups most affected by police shootings among the Black and White populations.



Smooth Approximation for White People:

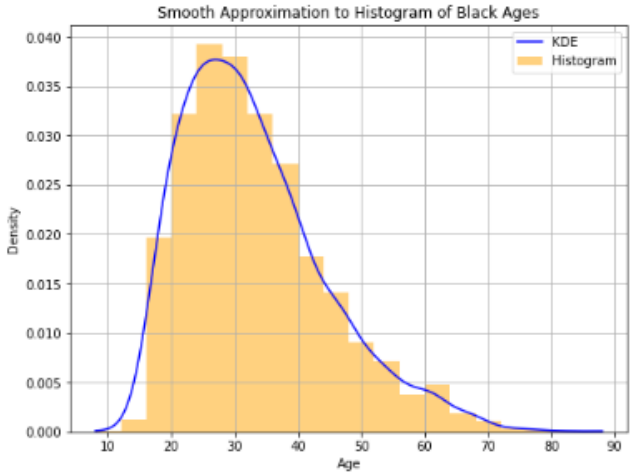
Smooth Approximation of white ages, from a dataset related to police shootings.




```
Minimum of white_ages = 8.0
Maximum of white_ages = 88.0
Median of white_ages = 31.0
Mean of white_ages = 33.04860442733398
Skewness of white_ages = 0.947861512161623
Kurtosis of white_ages = 0.8311188474405724
```

Smooth Approximation for Black People:

Smooth Approximation of white ages, from a dataset related to police shootings.




```
Minimum of black_ages = 8.0
Maximum of black_ages = 88.0
Median of black_ages = 31.0
Mean of black_ages = 33.04860442733398
Skewness of black_ages = 0.947861512161623
Kurtosis of black_ages = 0.8311188474405724
```

- **Monte-Carlo Method for Estimating The P-Value:**

```
Observed Mean Age Difference: -7.222270824175077
P-value (Monte Carlo): 0.0
Time taken to estimate the p-value using Monte-Carlo method is 1158.00627 seconds
```

The `P-value` of 0.0 indicates that the observed difference in means is statistically significant. It is performing statistical analysis found a significant difference between the two groups.

Appendix C: DATA AND CODE

- **Cumulative Distribution function:**

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

# Load the data from the CSV file
data = pd.read_csv('Police Shootings Age Race.csv')

# Separate data for black and white races
black_ages = list(data[data['race'] == 'B']['age'])
white_ages = list(data[data['race'] == 'W']['age'])

# Calculate cumulative distribution functions (CDFs)
black_cdf = np.cumsum(np.ones_like(black_ages)) / len(black_ages)
white_cdf = np.cumsum(np.ones_like(white_ages)) / len(white_ages)

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

plt.plot(black_ages, black_cdf, label='Black People', marker='o')
plt.plot(white_ages, white_cdf, label='White People', marker='o')

plt.title('Cumulative Distribution of Ages for Black and White People Shot by Police')
plt.xlabel('Age')
plt.ylabel('Cumulative Probability')
plt.grid(True)
plt.legend()

plt.show()

# Effect size calculation
effect_size = mean_black_age - mean_white_age
print("Mean age for black people shot by police:", mean_black_age)
print("Mean age for white people shot by police:", mean_white_age)
print("Effect size (Difference in mean ages):", effect_size)
```

- **Cohen's D:**

```
import pandas as pd
import numpy as np

# Load the data from the CSV file
data = pd.read_csv('Police Shootings Age Race.csv')

# Separate data for black and white races
black_ages = data[data['race'] == 'B']['age']
white_ages = data[data['race'] == 'W']['age']

# Calculate sample sizes
n1 = len(black_ages)
n2 = len(white_ages)

# Calculate sample variances
s1_sq = np.var(black_ages, ddof=1)
s2_sq = np.var(white_ages, 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(black_ages) - np.mean(white_ages)) / s_pooled

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

- **Mean Age Difference Between Black and White People:**

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

# Load the data from the CSV file
data = pd.read_csv('Police Shootings Age Race.csv')

# Separate data for black and white races
black_ages = data[data['race'] == 'B']['age']
white_ages = data[data['race'] == 'W']['age']

# Calculate the mean ages for black and white individuals
mean_black_age = np.mean(black_ages)
mean_white_age = np.mean(white_ages)

mean_difference_observed = mean_black_age - mean_white_age

# plot a histogram
plt.figure(figsize=(10, 6))
plt.hist(L,bins=15,density=True,alpha=0.7,color='skyblue', edgecolor='black')
plt.axvline(x=mean_difference_observed,color='red',linestyle='dashed',linewidth=2,label='Observed Mean Difference')
plt.title('Distribution of Mean Differences')
plt.xlabel('Mean Difference')
plt.ylabel('Density')
plt.legend()
plt.show()

print('Mean age difference for Black and White people =',mean_difference_observed)
num_samples_greater= np.sum(L>mean_difference_observed)
print('Number of Samples with mean difference > {:.5f}:'.format(mean_difference_observed),num_samples_greater)
# Calculate the p-value by comparing the observed difference to the differences obtained from permutations
p_value = np.mean([difference >= mean_difference_observed for difference in L])

# Print the p-value
print("P-value:", p_value)
```

Q-Q Plot of ages for Black people shot by police

```
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('Police Shootings Age Race.csv')

# Filter data for black people
black_ages = np.array(data[data['race'] == 'B']['age'])

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

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

plt.title('QQ Plot of Ages for Black People Shot by Police')
plt.xlabel('Theoretical Quantiles')
plt.ylabel('Sample Quantiles')
plt.grid(True)
plt.show()
```

Q-Q Plot of ages for White people shot by police

```
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('Police Shootings Age Race.csv')

# Filter data for black people
white_ages = np.array(data[data['race'] == 'W']['age'])

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

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

plt.title('QQ Plot of Ages for White People Shot by Police')
plt.xlabel('Theoretical Quantiles')
plt.ylabel('Sample Quantiles')
plt.grid(True)
plt.show()
```

Comparison of Distribution for Black and White Ages:

```
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('Police Shootings Age Race.csv')

# Filter data for black people
black_ages = np.array(data[data['race'] == 'B']['age'])

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

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

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

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

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

# Add labels and legend
plt.title('Smooth Approximation to Histogram of Black Ages')
plt.xlabel('Age')
plt.ylabel('Density')
plt.legend()

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

Smooth Approximation for White People:

```
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('Police Shootings Age Race.csv')

# Filter data for black people
white_ages = np.array(data[data['race'] == 'W']['age'])

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

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

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

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

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

# Add labels and legend
plt.title('Smooth Approximation to Histogram of White Ages')
plt.xlabel('Age')
plt.ylabel('Density')
plt.legend()

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

# Calculate statistical information for the 'pre_molt' column
white_ages_stats = data[data['race'] == 'B']['age'].describe()
white_ages_skewness = data[data['race'] == 'B']['age'].skew()
white_ages_kurtosis = data[data['race'] == 'B']['age'].kurtosis()

# Print the statistical information
print("Minimum of white_ages =", white_ages_stats['min'])
print("Maximum of white_ages =", white_ages_stats['max'])
print("Median of white_ages =", white_ages_stats['50%'])
print("Mean of white_ages =", white_ages_stats['mean'])
print("Skewness of white_ages =", white_ages_skewness)
print("Kurtosis of white_ages =", white_ages_kurtosis)

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

- **Monte-Carlo Method:**

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

# Load the data from the CSV file
data = pd.read_csv('Police Shootings Age Race.csv')

# Separate data for black and white races
black_ages = data[data['race'] == 'B']['age'].values
white_ages = data[data['race'] == 'W']['age'].values

# Calculate the observed mean age difference
mean_difference_observed = np.mean(black_ages) - np.mean(white_ages)

pooled_ages = np.concatenate([black_ages, white_ages])
np.random.shuffle(pooled_ages)


# Start timing
t1 = time.time()

# Number of Monte Carlo simulations
num_simulations = 10000000

# Initialize an array to store the simulated mean differences
simulated_differences = []

# Perform Monte Carlo simulations
for _ in range(num_simulations):
    # Generate random samples of black and white ages with replacement
    sampled_ages_1 = np.random.choice(pooled_ages, size=len(black_ages), replace=True)
    sampled_ages_2 = np.random.choice(pooled_ages, size=len(white_ages), replace=True)

    # Calculate the mean age difference for the random samples
    mean_difference = np.mean(sampled_ages_1) - np.mean(sampled_ages_2)
    simulated_differences.append(mean_difference)

# Calculate the p-value
p_value = np.mean([difference <= mean_difference_observed for difference in simulated_differences])

print("Observed Mean Age Difference:", mean_difference_observed)
print("P-value (Monte Carlo):", p_value)

# Print the time taken
print(f"Time taken to estimate the p-value using Monte-Carlo method is {time.time() - t1:.5f}
seconds")
```

Comparison of Distributions of Ages:

```
import numpy as np
import time
import matplotlib.pyplot as plt

# Load the data from the CSV file
data = pd.read_csv('Police Shootings Age Race.csv')

# Separate data for black and white races
black_ages = data[data['race'] == 'B']['age']
white_ages = data[data['race'] == 'W']['age']

# Start timing
t1 = time.time()

pooled_ages = np.concatenate([black_ages, white_ages])
t = np.sum(pooled_ages)

# Calculate the number of black and white individuals
nf = len(black_ages)
nw = len(white_ages)
L = []
# Number of iterations for permutation test
max_val = 2 * 10**6

# Perform permutation test
n = 1
while n < max_val:
    A = np.random.choice(pooled_ages, nw, replace=False)
    a = np.mean(A)
    b = np.mean((t - np.sum(A)) / nf)
    L.append(a - b)
    n += 1
# Convert the list to a numpy array
L = np.array(L).flatten()

# Print the time taken
print(f"Time taken to estimate the p-value using Monte-Carlo method is {time.time() - t1:.5f} seconds")
```

References:

[1] *MTH 522 (Advanced Mathematical Statistics, sections 02)*
https://mth522.wordpress.com/

[2] **https://www.dropbox.com/scl/fi/3ukmq88kh4uowplv8f6z7/Police-Shootings-Age-Race.csv?rlkey=cic8y3kcrgf0lq18jujsuek2d&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.
Roshni Pal: 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.
Trina Xavier: Worked on initial analyses, analyzing and looking for different models to fit the data on, testing various fits for their errors to describe trends between predictors.