Trinity College Dublin

The University of Dublin

Group Assignment 2

Fitting a probability distribution

Group 41:

Leo Lee

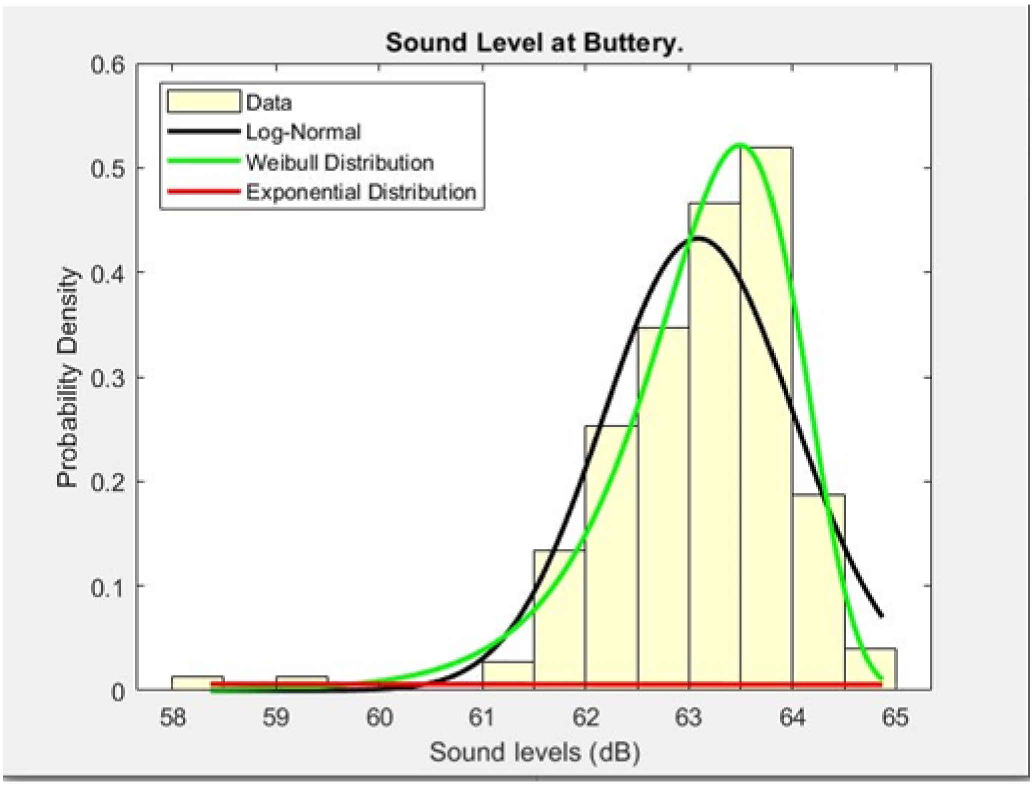
Lorraine Mathew

Sadbh O’Farrell

Victor Pinto Gomez De Zamora

Suhani Singla

Part A: Visualisation



-

Figure 1: Histogram with Fitted Distribution for the Buttery

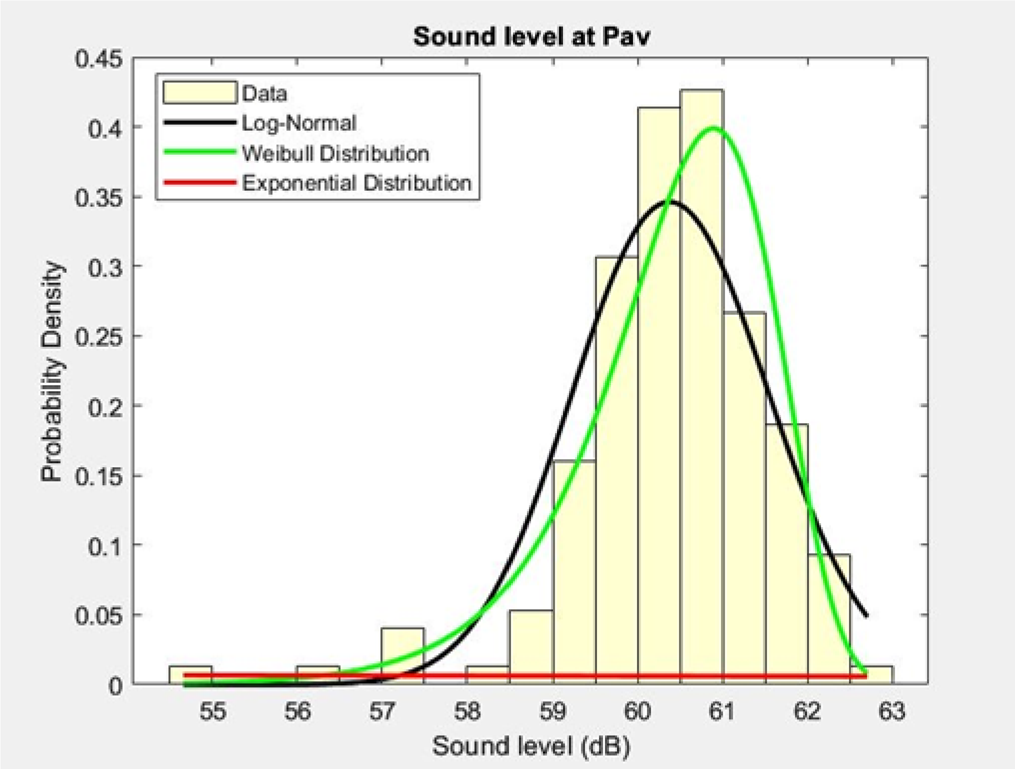


Figure 2: Histogram with Fitted Distribution for the Pav

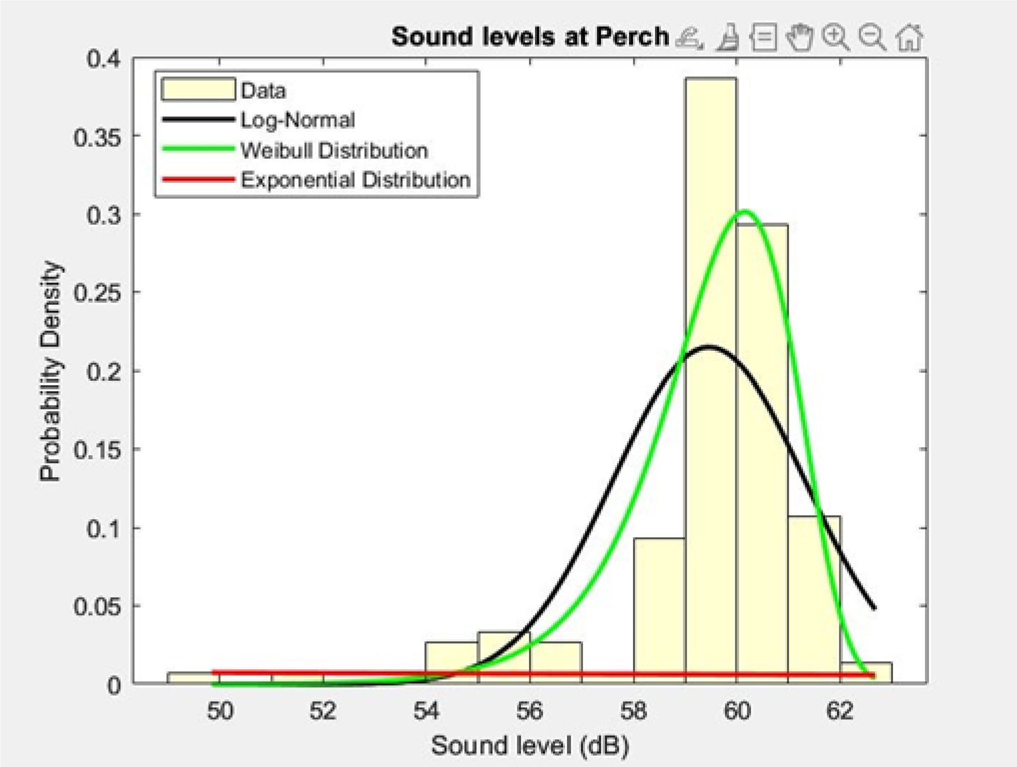
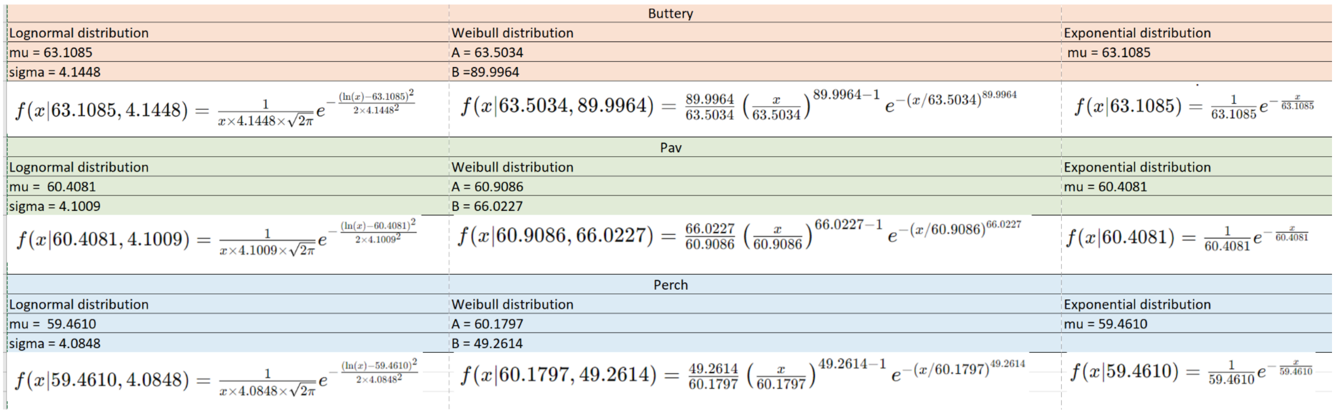


Figure 3: Histogram with Fitted Distribution for Butter

Part B: Parameter Estimation



	Buttery	
Lognormal distribution	Weibull distribution	Exponential distribution
mu = 63.1085	A = 63.5034	mu = 63.1085
sigma = 4.1448	B =89.9964	
$f(x\|63.1085, 4.1448) = \frac{1}{x \times 4.1448 \times \sqrt{2\pi}} e^{-\frac{(\ln(x) - 63.1085)^2}{2 \times 4.1448^2}}$	$f(x\|63.5034, 89.9964) = \frac{89.9964}{63.5034} \left(\frac{x}{63.5034}\right)^{89.9964-1} e^{-(x/63.5034)^{89.9964}}$	$f(x\|63.1085) = \frac{1}{63.1085} e^{-\frac{x}{63.1085}}$
	Pav	
Lognormal distribution	Weibull distribution	Exponential distribution
mu = 60.4081	A = 60.9086	mu = 60.4081
sigma = 4.1009	B = 66.0227	
$f(x\|60.4081, 4.1009) = \frac{1}{x \times 4.1009 \times \sqrt{2\pi}} e^{-\frac{(\ln(x) - 60.4081)^2}{2 \times 4.1009^2}}$	$f(x\|60.9086, 66.0227) = \frac{66.0227}{60.9086} \left(\frac{x}{60.9086}\right)^{66.0227-1} e^{-(x/60.9086)^{66.0227}}$	$f(x\|60.4081) = \frac{1}{60.4081} e^{-\frac{x}{60.4081}}$
	Perch	
Lognormal distribution	Weibull distribution	Exponential distribution
mu = 59.4610	A = 60.1797	mu = 59.4610
sigma = 4.0848	B = 49.2614	
$f(x\|59.4610, 4.0848) = \frac{1}{x \times 4.0848 \times \sqrt{2\pi}} e^{-\frac{(\ln(x) - 59.4610)^2}{2 \times 4.0848^2}}$	$f(x\|60.1797, 49.2614) = \frac{49.2614}{60.1797} \left(\frac{x}{60.1797}\right)^{49.2614-1} e^{-(x/60.1797)^{49.2614}}$	$f(x\|59.4610) = \frac{1}{59.4610} e^{-\frac{x}{59.4610}}$

Table 1: Estimated Parameters and PDFs

Part C: Goodness-of-Fit Comparison

	Buttery		**Pav**		**Perch**	
Log-normal	h=1	p=0.0011	h=0	p=0.1034	h=1	p=0.0000
Weibull	h=0	p=0.5148	h=1	p=0.0359	h=1	p=0.0000
Exponential	h=1	p=0.0000	h=1	p=0.0000	h=1	p=0.0000

Table 2: Initial Goodness-of-Fit Results

During the first test phase, it was discovered that the noise levels at the Buttery location were a good fit for the Weibull distribution, while the data from the Pav location best matched the log-normal distribution. However, neither of these distributions seemed to fit the Perch data well initially.

To further examine the Perch data, we used a boxplot analysis and found outliers in the plot. In order to improve the fit of the data, we decided to remove these extreme values and retest the distributions.

Goodness-of-Fit Results After Cleaning

Goodness-of-fit results for cleaned Perch data:

Cleaned Log-normal: h = 0, p = 0.2128

Cleaned Weibull: h = 1, p = 0.0005

Cleaned Exponential: h = 0, p = 0.1028

After identifying that there were some unusual data points that might be causing our test failures, we removed them and ran the goodness-of-fit analyses again. Following the clean-up, we made the following observations about the Perch data:

- The log-normal distribution now fits the data quite well, with a p-value of 0.2128, and is therefore a realistic model.
- Despite the removal of outliers, the Weibull distribution still did not fit the data well, as it had an extremely low p-value.
- The Exponential distribution, on the other hand, demonstrated a strong fit, with a p-value of 0.1028.

Final Observations

Our final look at the data suggests that after cleaning up the Perch dataset, the Log-normal distribution is a solid fit. Our initial findings remain valid for the Buttery and Pav locations. The Buttery data fits the Weibull distribution well, while the Pav data has the best fit with the Log-normal distribution.

Appendix

```
close all;

clear;

time = 1:1800;

% Read the Excel file into a matrix

dataMatrix = readmatrix('assignment_2_dataset.xlsx');

% Separate the columns into three different arrays

perch = dataMatrix(:, 1);  % First column

pav = dataMatrix(:, 2);  % Second column

buttery = dataMatrix(:, 3);  % Third column

%------------------------------------------------

% Plot histogram of sound levels in perch

figure(1)

histogram(perch, 'FaceColor', [1 1 0.7] , 'Normalization', 'pdf')

xlabel('Sound level (dB)')

ylabel('Probability Density')

title('Sound levels at Perch')

% Fit a log-normal distribution to the data and plot the curve

hold on

mu = mean(perch);

sigma = std(perch);

x_values = linspace(min(perch), max(perch), 1000);

y_values = normpdf(x_values, mu, sigma);

plot(x_values, y_values, 'k-', 'LineWidth', 2)

mu_log_pe = mean(perch);
```

```
sigma_log_pe = std(perch);
mean_data_pe = mean(perch);
var_data_pe = var(perch);
sigma_calc_pe = sqrt(log(1 + var_data_pe / mean_data_pe^2));
mu_calc_pe = log(mean_data_pe) - 0.5 * sigma_calc_pe^2;
% Remove non-positive values for Weibull fitting
perch_positive = perch(perch > 0);
pd_perch = fitdist(perch_positive, 'Weibull');
x_values = linspace(min(perch_positive), max(perch_positive), 1000);
y_values = pdf(pd_perch, x_values);
plot(x_values, y_values, 'g-', 'LineWidth', 2)
pd_weibull_pe = fitdist(perch_positive, 'Weibull');
a_weibull_pe = pd_weibull_pe.ParameterValues(1);  % Scale parameter
b_weibull_pe = pd_weibull_pe.ParameterValues(2);  % Shape parameter
%Exponential
perch_positive = perch(perch > 0);
pd_perch_exp = fitdist(perch_positive, 'exponential');
x_values = linspace(min(perch_positive), max(perch_positive), 1000);
y_values = pdf(pd_perch_exp, x_values);
plot(x_values, y_values, 'r-', 'LineWidth', 2)
pd_exponential_pe = fitdist(perch_positive, 'Exponential');
lambda_exponential_pe = pd_exponential_pe.ParameterValues(1);
hold off
legend('Data', 'Log-Normal', 'Weibull Distribution','Exponential Distribution')
%-----------------------------------------------
```

```
% Plot histogram of sound levels in pav
figure(2)
histogram(pav,  'FaceColor', [1 1 0.7],'Normalization', 'pdf')
xlabel('Sound level (dB)')
ylabel('Probability Density')
title('Sound level at Pav')
% Fit a log-normal distribution to the data and plot the curve
hold on
mu_pav = mean(log(pav));
sigma_pav = std(log(pav));
x_values_pav = linspace(min(pav), max(pav), 1800);
y_values_pav = lognpdf(x_values_pav, mu_pav, sigma_pav);
plot(x_values_pav, y_values_pav, 'k-', 'LineWidth', 2)
mu_log_pa = mean(pav);
sigma_log_pa = std(pav);
mean_data_pa = mean(pav);
var_data_pa = var(pav);
sigma_calc_pa = sqrt(log(1 + var_data_pa / mean_data_pa^2));
mu_calc_pa = log(mean_data_pa) - 0.5 * sigma_calc_pa^2;
% Remove non-positive values for Weibull fitting
pav_positive = pav(pav > 0);
pd_pav = fitdist(pav_positive, 'Weibull');
x_values = linspace(min(pav_positive), max(pav_positive), 1000);
y_values = pdf(pd_pav, x_values);
plot(x_values, y_values, 'g-', 'LineWidth', 2)
```

```
pd_weibull_pa = fitdist(pav_positive, 'Weibull');
a_weibull_pa = pd_weibull_pa.ParameterValues(1);  % Scale parameter
b_weibull_pa = pd_weibull_pa.ParameterValues(2);  % Shape parameter
%Exponential
pav_positive = pav(pav > 0);
pd_pav_exp = fitdist(pav_positive, 'exponential');
x_values = linspace(min(pav_positive), max(pav_positive), 1000);
y_values = pdf(pd_pav_exp, x_values);
plot(x_values, y_values, 'r-', 'LineWidth', 2)
pd_exponential_pa = fitdist(pav_positive, 'Exponential');
lambda_exponential_pa = pd_exponential_pa.ParameterValues(1);
hold off
legend('Data', 'Log-Normal', 'Weibull Distribution','Exponential Distribution')
%-------------------------------------------------
% Plot histogram of sound levels in buttery
figure(3)
histogram(buttery, 'FaceColor', [1 1 0.7],'Normalization', 'pdf')
xlabel('Sound levels (dB)')
ylabel('Probability Density')
title('Sound Level at Buttery.')
% Fit a log-normal distribution to the data and plot the curve
hold on
mu_buttery = mean(log(buttery));
sigma_buttery = std(log(buttery));
x_values_buttery = linspace(min(buttery), max(buttery), 1800);
```

```
y_values_buttery = lognpdf(x_values_buttery, mu_buttery, sigma_buttery);

plot(x_values_buttery, y_values_buttery, 'k-', 'LineWidth', 2)

mu_log_b = mean(buttery);

sigma_log_b = std(buttery);

mean_data_b = mean(buttery);

var_data_b = var(buttery);

sigma_calc_b = sqrt(log(1 + var_data_b / mean_data_b^2));

mu_calc_b = log(mean_data_b) - 0.5 * sigma_calc_b^2;

% Remove non-positive values for Weibull fitting

buttery_positive = buttery(buttery > 0);

pd_buttery = fitdist(buttery_positive, 'Weibull');

x_values = linspace(min(buttery_positive), max(buttery_positive), 1000);

y_values = pdf(pd_buttery, x_values);

plot(x_values, y_values, 'g-', 'LineWidth', 2)

pd_weibull_b = fitdist(buttery, 'Weibull');

a_weibull_b = pd_weibull_b.ParameterValues(1);  % Scale parameter

b_weibull_b = pd_weibull_b.ParameterValues(2);  % Shape parameter

%Exponential

buttery_positive = buttery(buttery > 0);

pd_buttery_exp = fitdist(buttery_positive, 'exponential');

x_values = linspace(min(buttery_positive), max(buttery_positive), 1000);

y_values = pdf(pd_buttery_exp, x_values);

plot(x_values, y_values, 'r-', 'LineWidth', 2)

pd_exponential_b = fitdist(buttery_positive, 'Exponential');

lambda_exponential_b = pd_exponential_b.ParameterValues(1);
```

```
hold off

legend('Data', 'Log-Normal', 'Weibull Distribution','Exponential Distribution')
```

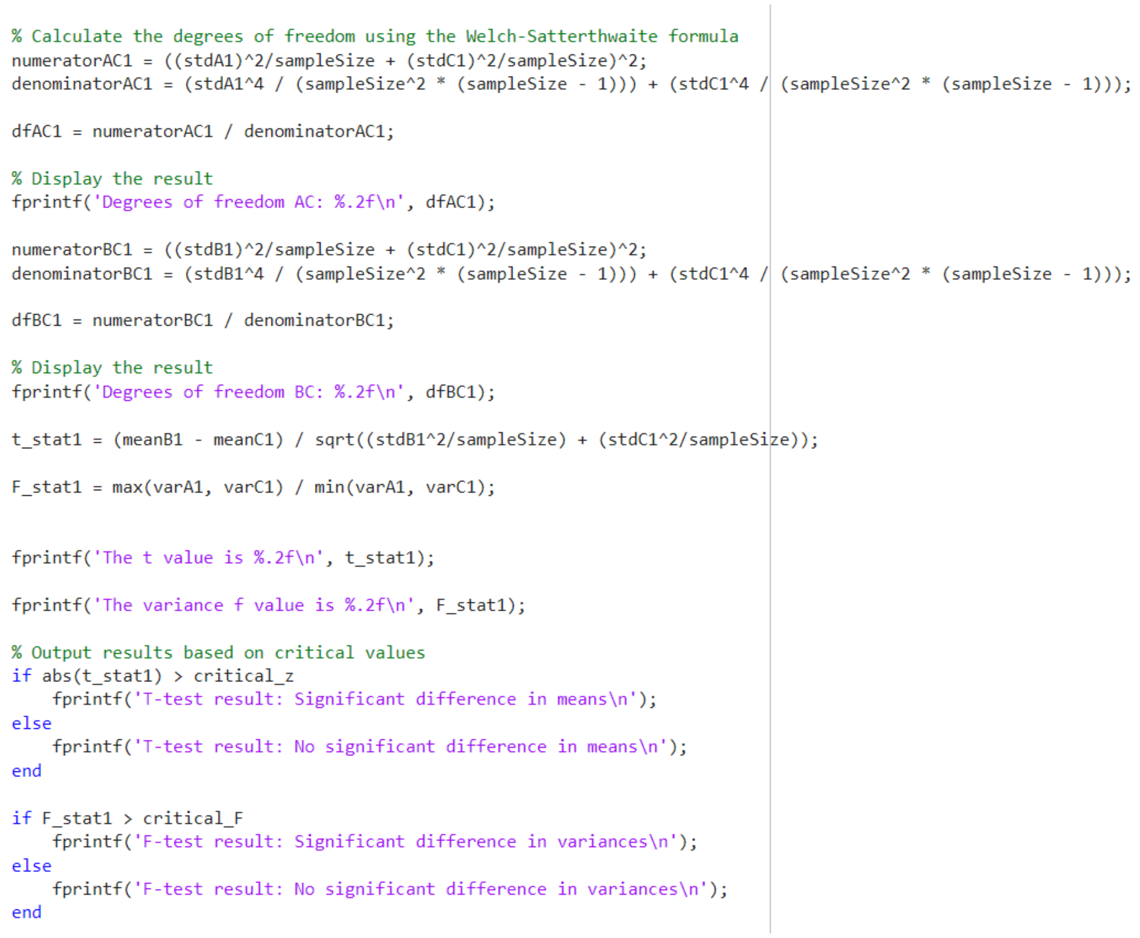
Group Assignment 3

Comparison of datasets

Group 41

This assignment required the group to compare the noise levels at different locations within a scenario between scenarios. The group started with 3 data sets from the previous assignment and another dataset from a different group. These were compiled to analyse the data appropriately.

To set up the hypothesis test the degree of freedom is calculated. The results indicated a high degree of freedom of over 1000 which allows for the assumption that the distribution of the data is normally distributed allowing for the usage of Z-scores for the 95% confidence interval, Z-score = 1.96. The calculations were done in matlab.



```
% Calculate the degrees of freedom using the Welch-Satterthwaite formula
numeratorAC1 = ((stdA1)^2/sampleSize + (stdC1)^2/sampleSize)^2;
denominatorAC1 = (stdA1^4 / (sampleSize^2 * (sampleSize - 1))) + (stdC1^4 / (sampleSize^2 * (sampleSize - 1)));

dfAC1 = numeratorAC1 / denominatorAC1;

% Display the result
fprintf('Degrees of freedom AC: %.2f\n', dfAC1);

numeratorBC1 = ((stdB1)^2/sampleSize + (stdC1)^2/sampleSize)^2;
denominatorBC1 = (stdB1^4 / (sampleSize^2 * (sampleSize - 1))) + (stdC1^4 / (sampleSize^2 * (sampleSize - 1)));

dfBC1 = numeratorBC1 / denominatorBC1;

% Display the result
fprintf('Degrees of freedom BC: %.2f\n', dfBC1);

t_stat1 = (meanB1 - meanC1) / sqrt((stdB1^2/sampleSize) + (stdC1^2/sampleSize));

F_stat1 = max(varA1, varC1) / min(varA1, varC1);


fprintf('The t value is %.2f\n', t_stat1);

fprintf('The variance f value is %.2f\n', F_stat1);

% Output results based on critical values
if abs(t_stat1) > critical_z
    fprintf('T-test result: Significant difference in means\n');
else
    fprintf('T-test result: No significant difference in means\n');
end

if F_stat1 > critical_F
    fprintf('F-test result: Significant difference in variances\n');
else
    fprintf('F-test result: No significant difference in variances\n');
end
```

Location comparison:

B and C/A and C:

H_0: $\mu_B = \mu_c$

H_1: $\mu_B \neq \mu_c$

Dining Hall:

The mean value at location B was 59.64 while the mean value at location C was 69.64. The mean value at location A was 63.11. The best way to analyse these figures is by using a t-test or an F-test. The results for which are displayed in the appendix below. The t-test is a statistical method used to assess whether the means of two independent samples are significantly different from each other. In this context, it helps us understand if the noise levels at locations A, B and C vary significantly or are essentially the same. The F-test helps us understand if the noise levels exhibit similar levels of variability between the two locations.

The t-test revealed that there is no significant difference between the mean noise levels at these locations. As the t-value is relatively low, it reinforces the conclusion that the noise levels at locations A,B and C are comparable. The f-test indicates that both the mean noise levels and the variability of noise levels at locations B and C are not statistically different. Therefore we fail to reject the null hypothesis.

Gates

The mean noise levels at locations A, B, and C were 62.92, 59.01, and 61.55, respectively. To analyze these values, we utilized both a t-test and an F-test again. The results of the t-test revealed no significant difference between the mean noise levels at these locations. Additionally, the F-test indicated that both the mean noise levels and the variability of noise levels at locations A,B and C are not statistically different. Therefore we fail to reject the null hypothesis.

Therefore, based on the mean noise levels provided and supported by the statistical analyses, the dining hall (location C) appears to be the noisiest among the locations considered.

Step 3: Scenario comparison

As the overall mean noise level for the dining hall area (64.13) is higher than that of the gates area (61.16), it suggests that the dining hall area tends to be noisier than the campus gates. This comparison is based on the provided mean noise levels for each area.

The lack of significant differences in mean noise levels between locations B and C within both the dining hall and gates areas suggests that noise levels are relatively consistent within each area.

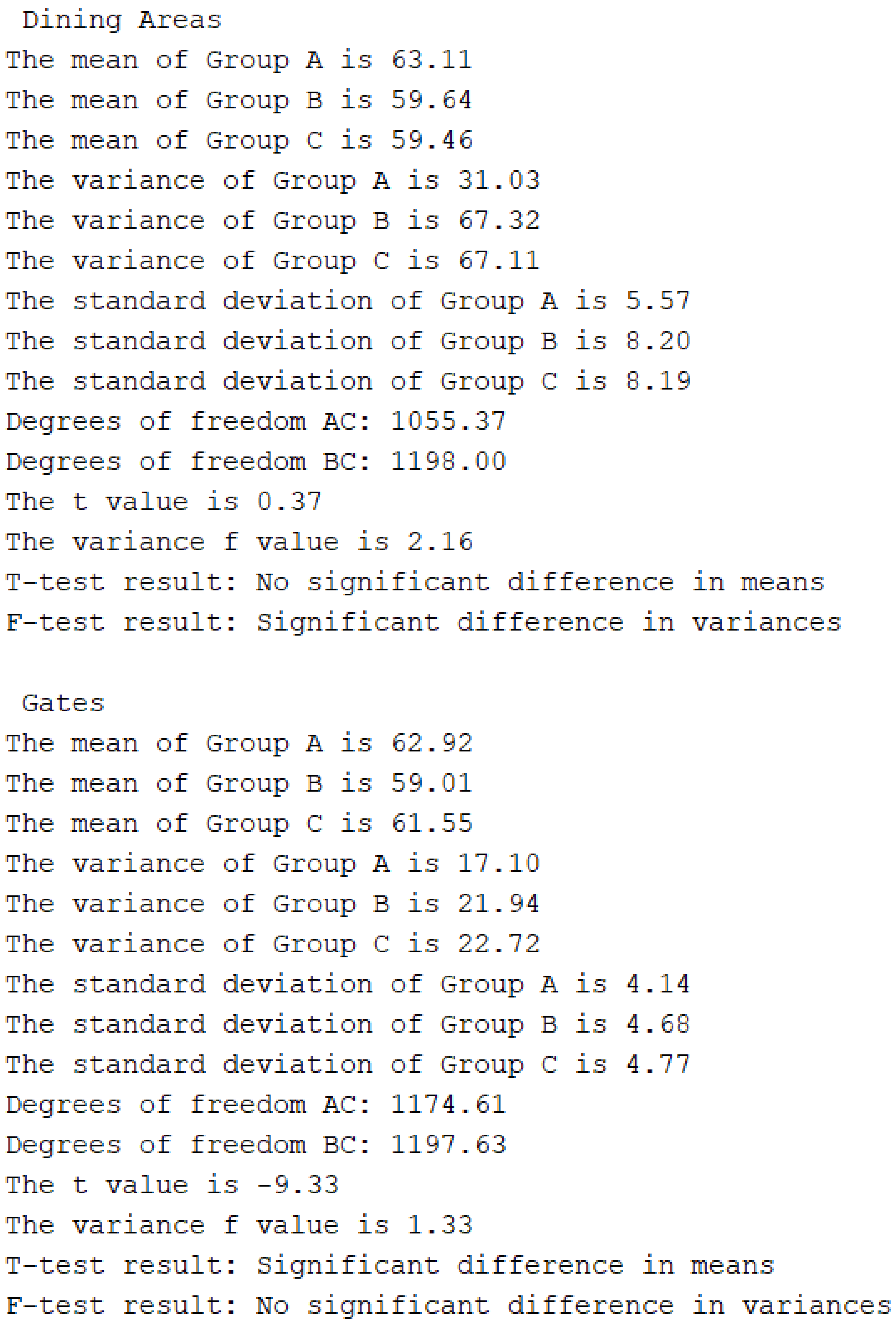
Conclusion:

In conclusion, the analysis of noise levels across different locations within the scenario reveals valuable insights. Through statistical testing, it is evident that there are no significant differences in noise levels between locations A, B, and C within both the dining hall and gates areas. This suggests a consistency in noise levels within each area.

Further examination comparing the dining hall and gates areas indicates that the dining hall tends to have higher noise levels, as evidenced by its overall mean noise level being greater than that of the gates area. This observation is supported by statistical analyses using t-tests and F-tests.

Overall, the findings suggest that the dining hall, specifically location C, tends to be the noisiest among the locations considered. This analysis aids in understanding the noise distribution within different areas of the scenario, providing valuable insights for potential noise management strategies or further investigations into the sources of noise.

Appendix:



```
 Dining Areas
The mean of Group A is 63.11
The mean of Group B is 59.64
The mean of Group C is 59.46
The variance of Group A is 31.03
The variance of Group B is 67.32
The variance of Group C is 67.11
The standard deviation of Group A is 5.57
The standard deviation of Group B is 8.20
The standard deviation of Group C is 8.19
Degrees of freedom AC: 1055.37
Degrees of freedom BC: 1198.00
The t value is 0.37
The variance f value is 2.16
T-test result: No significant difference in means
F-test result: Significant difference in variances

 Gates
The mean of Group A is 62.92
The mean of Group B is 59.01
The mean of Group C is 61.55
The variance of Group A is 17.10
The variance of Group B is 21.94
The variance of Group C is 22.72
The standard deviation of Group A is 4.14
The standard deviation of Group B is 4.68
The standard deviation of Group C is 4.77
Degrees of freedom AC: 1174.61
Degrees of freedom BC: 1197.63
The t value is -9.33
The variance f value is 1.33
T-test result: Significant difference in means
F-test result: No significant difference in variances
```

All results