University of Glasgow

VIA VERITAS VITA

**<u>Assessed Exercise Report</u>**

**Computational Social Intelligence - COMPSCI 4080**

**Professor Name: Dr. Alessandro Vinciarelli**

**Student Name: Omar Jabri**

**Student Number: 2519359J**

# Part one:

## I- Introduction:

This report describes the process taken to perform several statistical tests on a set of data extracted from socially convicted events. The latter is a collection of laughter events, said to be a nonverbal cue, extracted during 60 phone calls between 120 speakers. A nonverbal cue is “the process of one person simulating meaning in the mind of another person or persons by means of nonverbal messages” (Richmond et al., 1991) [1].
By analyzing the dataset, the resulting properties were found:

1- Every laughter event contains gender and role of person that laughs.
2- Every laughter event contains the duration of the laughter in seconds.

The distribution of the duration of calls for both genders Male and Female as well as associated with Role i.e. Caller and Receiver, can be found in the following histogram:

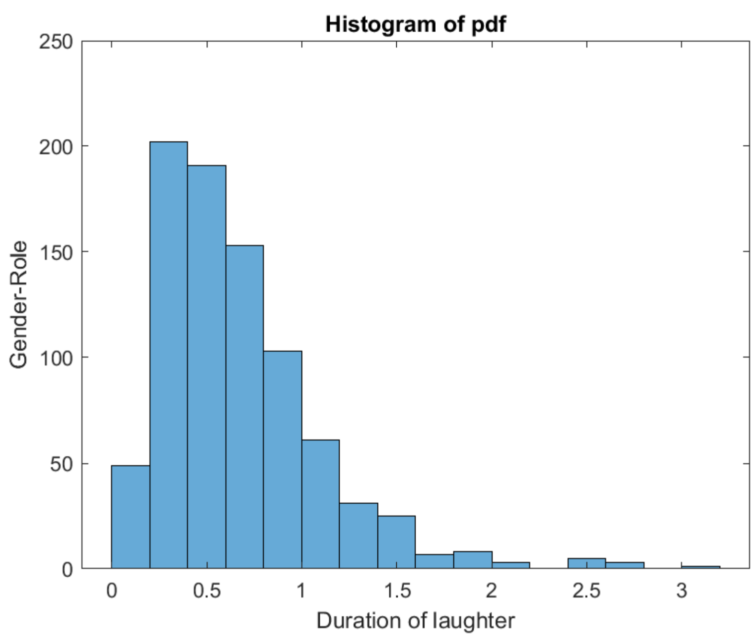



*Figure 1: Histogram of distribution of duration of call across Gender/Role class*

Moreover, the data consists of 120 speakers, split evenly between callers and receivers i.e. (60,60), and consisting of 57 female and 63 male speakers. The total dataset shows some statistical properties. The mean of the duration of calls irrespective of gender or role is of 0.6671 seconds. The standard deviation of the latter is of 0.4296 seconds. In addition, the longest duration of laughter events is of 3.139 seconds, whilst the shortest duration is of 0.061 seconds.

In the following sections, several statistical tests are performed to answer the questions below:

1- Is the number of laughter events higher for women than for men?
2- Is the number of laughter events higher for callers than for receivers?
3- Are laughter events longer for women?
4- Are laughter events longer for callers?

## II- <u>Methodology:</u>

**<u>Problem 1:</u>** Is the number of laughter events higher for women than for men?

From the question posed above, the research hypothesis and null hypothesis can be extracted:
-**Research Hypothesis**: Women tend to laugh more than men during phone calls.
-**Null Hypothesis**: There is no difference between the number of laughter events for women and men.
Before tackling the problem at hand, an analysis of the data and its statistics has been implemented for future tests.
Histograms illustrating the distribution of duration of laughter events for female and male respectively are illustrated below:

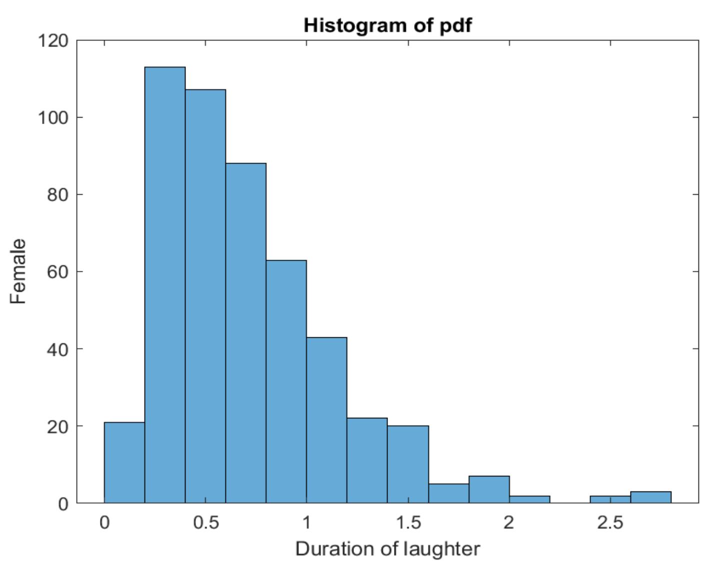



*Figure 2: Histogram of distribution of duration of call across Female class*

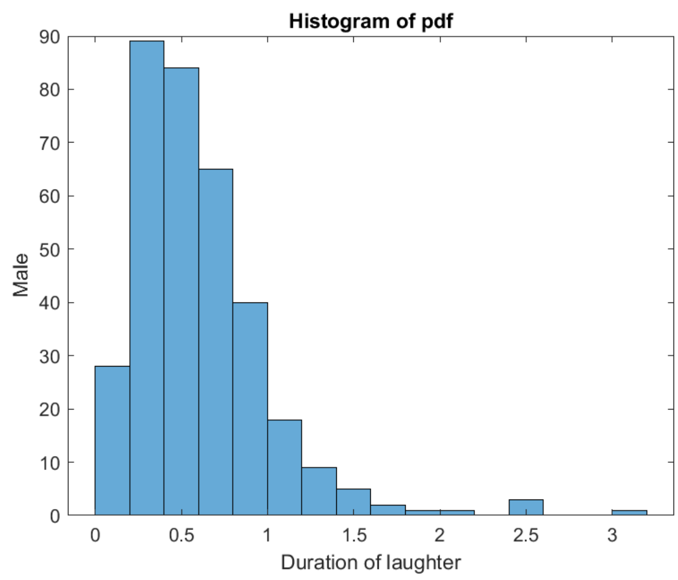



*Figure 3: Histogram of distribution of duration of call across Male class*

Moreover, some important statistics have been calculated before performing any statistical test.

| Gender | Count | Variance | Mean |
|---|---|---|---|
| Female | 496 | 0.19502 | 0.7096 |
| Male | 346 | 0.16375 | 0.6062 |

This question relates the number of times laughter events happens to the gender of that person. Hence, the values to be tested are discrete. Moreover, the test used will result in a decision of either rejecting the Null Hypothesis $H_0$ or failing to reject it (i.e. accepting the Research Hypothesis $H_1$). Thus, the **One Tailed Goodness of fit Chi Square** test has been chosen. A one-tailed test has been chosen merely since the test is performed for the possibility of the relationship between number of laughter events of both women and men in one direction. This will provide more power in rejecting the null Hypothesis. The significance level of a one tailed test is usually equal to 0.05 (value chosen throughout this report), i.e. 5% region of the probability distribution. Any calculated statistic used, such as the mean, or in our case the Chi square statistic $X^2$, that lies beyond or on this 5% region, will lead to a rejection of the Null Hypothesis. In more detail, this region is called the rejection region, where $P(X^2 > X^2_\alpha)$.

The Chi Square formula is as follows: $X^2 = \sum \frac{(O-E)^2}{E}$ (1).

This equation (1) has been developed from scratch using MATLAB (check Appendix A).

The Chi Square is used as an approach to test if there is a statistically significant difference between the observed values of number of times laughter events happened and the expected values for both women and men. Since there are only two observations, the degrees of freedom of this statistic is calculated as follows:

Degrees of freedom = Number of observations – 1 = 2 – 1 = 1.

Having the significance level and degrees of freedom set, the next task is to determine the observed and expected values.

For both females and males, the observed values are simply the count of the number of times laughter events happened for women and men. The latter are 496 and 346, as illustrated in the table above. However, the expected values are calculated by considering the number of speakers for both genders:

$E_F = C*(F/S)$ (2). Where $E_F$ is the expected value for number of times laughter event happened for Female, C is the count of all laughter events, F is the number of Female speakers and S is the number of total speakers.

By plugging in the numbers in equation (2), $E_F$ becomes = 842*(57/120) = 442.05.

This procedure is as well implemented to calculate $E_m$ (expected value for number of times laughter event happened for Male). This results in $E_m$ = 399.95.

Having both the observed and expected values calculated, they can therefore be plugged in the chi square formula (1) to find the Chi Square statistic and infer whether the Null Hypothesis can be rejected or not.

The resulting Chi Square statistic is $X^2_{calculated} = 13.8617$.

The only task left to do is to compare the calculated $X^2$ with the one at the significance level of 0.05. The latter is $X^2_{0.05} = 3.8145$.

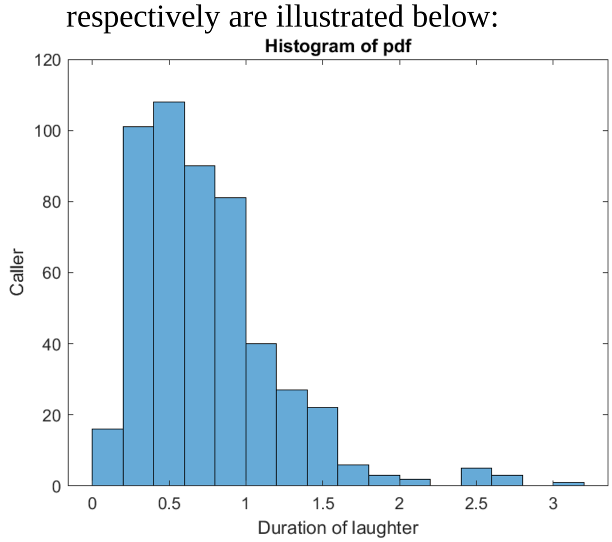
**Since $X^2_{calculated} > X^2_{0.05}$, the Null Hypothesis is rejected, which confirms that the number of laughter events is higher for women. Hence, there is a statistically significant difference between the number of laughter events for women and men.**

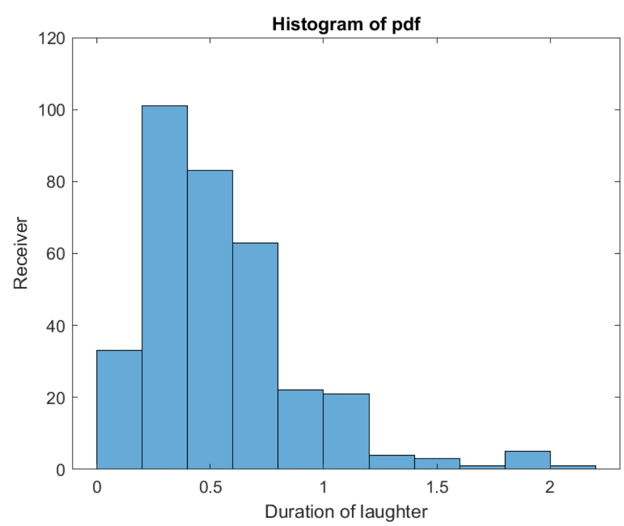
All calculations have been implemented using MATLAB. Chi square value has been extracted by a MATLAB function written from scratch. Finding the Chi Square value at the significance level was implemented by generating the whole distribution and extracting the statistic at 0.05 significance level and with degrees of freedom of value 1. Functions and code related to this part of the exercise can be found in Appendix A.

**Problem 2:** Is the number of laughter events higher for callers than for receivers?

**Note:** This solution follows the exact procedure of problem 1.

From the question posed above, the research hypothesis and null hypothesis can be extracted:
-**Research Hypothesis**: Callers tend to laugh more than receivers during phone calls.
-**Null Hypothesis**: There is no difference between the number of laughter events for callers and receivers.
Before tackling the problem at hand, an analysis of the data and its statistics has been implemented for future tests.
Histograms illustrating the distribution of duration of laughter events for female and male respectively are illustrated below:



.

Some important statistics have been calculated before performing any statistical test

| Role | Count | Variance | Mean |
|---|---|---|---|
| Callers | 505 | 0.2126 | 0.7457 |
| Receivers | 337 | 0.1197 | 0.5494 |

This question relates the number of times laughter events happens to the role of that person. Hence, the values to be tested are discrete. Moreover, the test used will result in a decision of either rejecting the Null Hypothesis $H_0$ or failing to reject it (i.e. accepting the Research Hypothesis $H_1$). Thus, the **One Tailed Goodness of fit Chi Square** test has been chosen. A one-tailed test has been chosen merely since the test is performed for the possibility of the relationship between number of laughter events of both callers and receivers in one direction. This will provide more power in rejecting the null Hypothesis. The significance level of a one tailed test is usually equal to 0.05 (value chosen throughout this report), i.e. 5% region of the probability distribution. Any calculated statistic used, such as the mean, or in our case the Chi square statistic $X^2$, that lies beyond or on this 5% region, will lead to a rejection of the Null Hypothesis. In more detail, this region is called the rejection region, where $P(X^2 > X^2_\alpha)$.

The Chi Square formula is as follows: $X^2 = \sum \frac{(O-E)^2}{E}$ (3).

The Chi Square is used as an approach to test if there is a statistically significant difference between the observed values of number of times laughter events happened and the expected values for both callers and receivers. Since there are only two observations, the degrees of freedom of this statistic is calculated as follows:

Degrees of freedom = Number of observations – 1 = 2 – 1 = 1.

Having the significance level and degrees of freedom set, the next task is to determine the observed and expected values.

For both callers and receivers, the observed values are simply the count of the number of times laughter events happened for callers and receivers. The latter are 505 and 337, as illustrated in the table above. However, the expected values are calculated by considering the number of speakers for both roles:

$E_F = C*(Ca/S)$ (4)

Where $E_c$ is the expected value for number of times laughter event happened for Callers, C is the count of all laughter events, Ca is the number of Callers speakers and S is the number of total speakers.

By plugging in the numbers in equation (4), $E_c$ becomes = 842*(60/120) = 421.

This procedure is as well implemented to calculate $E_r$ (expected value for number of times laughter event happened for Receiver). This results in $E_r$ = 421.

Having both the observed and expected values calculated, they can therefore be plugged in the chi square formula (3) to find the Chi Square statistic and infer whether the Null Hypothesis can be rejected or not.

The resulting Chi Square statistic is $X^2_{calculated} = 33.5201$.

The only task left to do is to compare the calculated $X^2$ with the one at the significance level of 0.05. The latter is $X^2_{0.05} = 3.8145$.

**Since $X^2_{calculated} > X^2_{0.05}$, the Null Hypothesis is rejected, which confirms that the number of laughter events is higher for women. Hence, there is a statistically significant difference between the number of laughter events for callers and receivers.**

All calculations have been implemented using MATLAB. Chi square value has been extracted by a MATLAB function written from scratch. Finding the Chi Square value at the significance level was implemented by generating the whole distribution and extracting the statistic at 0.05 significance level and with degrees of freedom of value 1. Functions and code related to this part of the exercise can be found in Appendix A.

**Problem 3:** Are laughter events longer for women?

From the question posed above, the research hypothesis and null hypothesis can be extracted:
**-Research Hypothesis**: Women tend to laugh for longer durations during phone calls than men.
**-Null Hypothesis**: There is no difference between the mean of duration of laughter events for women and men.

In the questions posed above, the values to be tested on are continuous since time is a continuous variable. Hence, the correct test to perform is independent two sample Student t's test. In this question, the Null hypothesis considers that mean of duration for female – mean of duration of male = 0. However, the research hypothesis dictates that there is a statistically significant difference between the mean of durations, hence $\mu_F - \mu_M$ does not equal to zero. A two tailed test is then used, to test if $\mu_F$ is less than or greater than $\mu_M$. Moreover, there is no prior knowledge of the problem at hand (i.e. there is no prior knowledge about the general duration of laughter for women and men). Hence, this test cannot be reduced to a one-tailed test. Since there is no prior knowledge of the variance of and the mean of the populations, t test is calculated by using sample variances and sample means of the two groups. t test is calculated as follows:

$$t = \frac{x1 - x2}{\sqrt{var1^2 + var2^2}} \quad (5)$$

where x1 is the sample mean for Female, x2 is the sample mean of Male, $var_1^2 = s^2_1/n1$, which is the sample variance for group Female, and $var_1^2 = s^2_2/n2$ which is the sample variance for group Male. This equation (5) has been developed from scratch using MATLAB (check Appendix A).
Moreover, n1 and n2 are the group sizes for Female and Male, respectively.
In order to calculate the t statistic, the sample means, and variances have been extracted for both groups. The results are shown in a table below:

| Gender | Sample Mean | Sample Variance |
|---|---|---|
| Female | 0.7096 | 0.0198^2 |
| Male | 0.6062 | 0.0217^2 |

By plugging in these statistics in equation (5), the t test is calculated and results in a value of 3.5145. The degrees of freedom using t as a statistic are calculated by the following equation:
(n1 – 1) + (n2 – 1) = (496 - 1) + (346 - 1) = 840 degrees of freedom, where n1 and n2 are the sample sizes of groups Female and Male respectively.

Moreover, since this is a two tailed test, the significance level is equal to 0.05/2 = 0.025, to cover both regions of the probability distribution (hence why it is called a **two-tailed** test). In addition, any statistic (in this case the Students' t) that falls into either of these regions will lead to the rejection of the Null Hypothesis. In more detail, this region is called the rejection region, where $P(t > t_\alpha)$.

Having both degrees of freedom and significance level, the t at the critical value 0.025 can be extracted. The latter is: $t_{0.025} = 1.9627$.
Therefore, since $t_{calculated} > t_{0.025}$ (3.5145 > 1.9627), the Null Hypothesis can be rejected. Thus, it can be confirmed that women tend to laugher for longer durations than men. Hence, there is a statistically significant difference between the mean of duration of laughter events of groups Female and Male.

All calculations have been implemented using MATLAB. The t value has been extracted by a MATLAB function written from scratch. Finding the t value at the significance level was implemented by generating the whole distribution and extracting the statistic at 0.025 significance level and with degrees of freedom of value 840. Functions and code related to this part of the exercise can be found in Appendix A.

**Problem 4:** Are laughter events longer for callers?

**Note:** This solution follows the exact procedure of problem 3.

From the question posed above, the research hypothesis and null hypothesis can be extracted:
-**Research Hypothesis**: Callers tend to laugh for longer durations during phone calls than receivers.
-**Null Hypothesis**: There is no difference between the mean of duration of laughter events for callers and receivers.

In the questions posed above, the values to be tested on are continuous since time is a continuous variable. Hence, the correct test to perform is independent two sample Student t's test. In this question, the Null hypothesis considers that mean of duration for caller – mean of duration of receiver = 0. However, the research hypothesis dictates that there is a statistically significant difference between the mean of durations, hence $\mu_C$ - $\mu_R$ does not equal to zero. A two tailed test is then used, to test if $\mu_C$ is less than or greater than $\mu_R$. Moreover, there is no prior knowledge of the problem at hand (i.e. there is no prior knowledge about the general duration of laughter for callers and receivers). Hence, this test cannot be reduced to a one-tailed test. Since there is no prior knowledge of the variance of and the mean of the populations, t test is calculated by using sample variances and sample means of the two groups. t test is calculated as follows:

$$t = \frac{x1 - x2}{\sqrt{var1^2 + var2^2}} \quad (6)$$

where x1 is the sample mean for Caller, x2 is the sample mean of Receivers, $var_1^2 = s^2{}_1/n1$, which is the sample variance for group Callers, and $var_1^2 = s^2{}_2/n2$ which is the sample variance for group Receivers.
Moreover, n1 and n2 are the group sizes for Callers and Receivers, respectively.
In order to calculate the t statistic, the sample means, and variances have been extracted for both groups. The results are shown in a table below:

| Role | Sample Mean | Sample Variance |
|---|---|---|
| Callers | 0.7457 | 0.0205^2 |
| Receivers | 0.5494 | 0.0188^2 |

By plugging in these statistics in equation (6), the t test is calculated and results in a value of 7.0469. The degrees of freedom using t as a statistic are calculated by the following equation:
(n1 – 1) + (n2 – 1) = (505 - 1) + (337 - 1) = 840 degrees of freedom, where n1 and n2 are the sample sizes of groups Callers and Receivers respectively.

Moreover, since this is a two tailed test, the significance level is equal to 0.05/2 = 0.025, to cover both regions of the probability distribution (hence why it is called a **two-tailed** test). In addition, any statistic (in this case the Students' t) that falls into either of these regions will lead to the rejection of the Null Hypothesis. In more detail, this region is called the rejection region, where $P(t > t_{\alpha})$.

Having both degrees of freedom and significance level, the t at the critical value 0.025 can be extracted. The latter is: $t_{0.025} = 1.9627$.
Therefore, since $t_{calculated} > t_{0.025}$ (7.0649 > 1.9627), the Null Hypothesis can be rejected. Thus, it can be confirmed that callers tend to laugher for longer durations than receivers.
Hence, there is a statistically significant difference between the mean of duration of laughter events of groups Callers and Receivers.

All calculations have been implemented using MATLAB. The t value has been extracted by a MATLAB function written from scratch. Finding the t value at the significance level was implemented by generating the whole distribution and extracting the statistic at 0.025 significance level and with degrees of freedom of value 840. Functions and code related to this part of the exercise can be found in Appendix A.

## III- Conclusion:

This assessed exercise aimed at implementing four statistical tests on four different problems. The first two problems were tackled by using a one-tailed Chi Square test which considers discrete variables (count of laughter events for the pairs female, male and caller, receiver). Both Null Hypotheses presented by the two problems have been rejected, which leads to a conclusion that women/callers tend to laugh more than men/receivers. The following two problems were addressed by using a two-tailed independent two sample Students' t test which considers continuous variables (duration of laughter events). Both Null Hypotheses presented by the two problems have also been rejected. Hence, women/callers tend to laugh for a longer duration than men/receivers. In hypothesis testing, reaching a validation of research hypothesis leads to inferring about future Hypotheses. One possible hypothesis would be that men and women behave as they have different status (in this study social stats), where men tend to feel like they have higher status, which will result in women laughing more and longer according to Vinciarelli (2015)[3] in his article "When the words are not everything: the use of laughter, fillers, back-channel, silence and overlapping speech in phone calls". Moreover, with respect to caller/receiver relations: "In the case of laughter, a possible explanation is that the cue is important in social discourse" (Provine and Young, 1991) [4].

## **Part two**:

### I- **Introduction**:

This part of the exercise describes the process of using Gaussian Discriminant Functions based classifier to classify feature vectors. The latter are vectors of action units on the face of a person. These feature vectors include 17 different action units. The classes are "smile" and "frown" and are represented as a categorical vector.

### II- **Facial Expression Analysis**:

According to Tian Y. et al, "Facial expressions are the facial changes in response to a person's internal emotional states, intentions, or social communications" [5].

Facial expressions can dictate a lot of information about the person behind that face. The analysis of these expressions would lead to reasoning about the emotions, intentions, etc. Humans communicate their emotions and intentions to each other by just changing their facial expressions and shaping their facial features [7]. With the advancements in technology, "much progress has been made to build computer systems to understand and use this natural form of human communication" (Tian Y. et al, 2001).

Facial expression analysis is clearly described by "describing facial movement based on an anatomical analysis of facial action", according to Ekman and Friesen (1976). Facial expression analysis is a field of study that analyzes the face of a human person and infers information from that analysis. This exercise aims at predicting the facial expression (either smiling or frowning) of a person using action units of the face as feature vectors.

One approach into solving this problem is proposed in Ekman and Friesen's paper "Measuring Facial Movement" (1976), where a "comprehensive system which could distinguish all possible visually distinguishable facial movements" called the Facial

Action Code is implemented. The latter is a system that measures movement of the face as stated by Ekman and Friesen (1976). This FAC system uses the action units described above to infer facial expressions “Our first step in developing FAC was to study various anatomical texts to discover the minimal units” [6].

Moreover, “FAC allows for measuring facial asymmetries, when different action units appear on each side of the face” [6]. Action units show the different facial muscles movements [8], i.e. they correspond to the activation of one or more facial muscles. Another approach to facial expression analysis is OpenFace. An open source tool that can perform facial expression analysis based on action units [9].
In this report, the process of predicting the facial expression using action units as feature vectors results in using Gaussian Discriminant Functions based Classifiers.

## III- **Methodology**:

### 1- **Theory:**

The following classifier is trained on a training dataset “training-part-2.csv” that includes 36 training examples. After training and learning the needed parameters, the classifier is tested on a test set “test-part-2.csv” that includes 16 test examples, where the error rate, i.e. the percentage of times the model classified the feature vector in the wrong class, is extracted. Since there are two classes to predict, this process is said to be a Binary Classification. The theory behind this problem can be introduced by going back to Probability theory.
Let x = feature and C = class, then the probability of classifying x as C is:

$$\mathrm{P(C|x)} = \frac{\mathrm{P(x|C)P(C)}}{P(x)} \quad (1)$$

This rule is known as Bayes rule and plays an important role in the nature of the classifier used in this exercise, the Naïve Base Classifier.
The latter is a classifier that tries to minimize the loss when it comes to classifying x in the wrong class. Hence:

$$\alpha^* = argmin_\alpha(P(\alpha(x) \neq C))$$

Where $\alpha^*$ is the Bayes classifier [10], $\alpha(x)$ is the classification of feature x, and C is the correct class.
This equation describes how the Bayes classifier tries to minimize this loss, leading to more accuracy in classifying features.
Since this is a Binary Classification, there are two classes $C_1$ = “smile” and $C_2$ = “frown” to be assigned to the features. These classes are mutually exclusive.

The feature vectors contain 17 action units:
[AU01,AU04,AU05,AU06,AU07,AU09,AU10,AU12,AU14,AU15,AU17,AU20,AU23,AU25,AU26,AU42].

A more accurate representation of a loss function would be the expected/conditional risk of taking action $\beta i$ [10]:

$$R(\beta i|x) = \sum_{j=1}^{M} \pi\left(\beta i \middle| C_j\right) p(C_j|x) \qquad (2)$$

Where $\beta i$ is the action taken in the classification and $\pi\left(\beta i \middle| C_j\right)$ is the loss due to the action [10].
This classifier follows a procedure called Bayes Decision Rule: “To minimize the overall risk, compute the conditional risk” [10].
This statement is translated in mathematical form by equation (2).
Hence, for the classifier to become more accurate, it needs to minimize the expected risk described in equation (2).
This will result in the classifier minimizing 1 – $P(C_i|x)$, hence maximizing $P(C_i|x)$.
This relation makes sense because choosing the class with highest likelihood will result in having a higher probability of classifying the feature correctly.
What is remaining is the usage of a discriminant function to map feature x to a certain class C. A popular discriminant function is of the shape:

$$\gamma_i(x) = -R(\beta_i|x)[10]$$

This results in minimizing the conditional risk by choosing the maximum $\gamma_i(x)$ [10].
Hence, the discriminant function becomes:

$$\gamma_i(x) = P(C_i|x) \text{ [10]}$$

As noted in equation 1, the discriminant function then becomes:

$$\gamma_i(x) = \frac{P(x|C_i)P(C_i)}{P(x)} \quad (3)$$

Where $C_i$ is a specific class i.
Equation (3) can be simplified by taking the natural logarithm on both sides [10].
This results in a final discriminant function form that is used throughout this exercise:

$$\gamma_i'(x) = \ln\left(P(x|C_i)\right) + \ln\left(P(C_i)\right) \quad (4)$$

Since the logarithm is monotonic, and that P(x) is constant for all, equation (4) holds.

Where $P(x|C_i)$ is the likelihood of classifying x in class $C_i$, and $P(C_i)$ is the prior probability.

In this exercise, Multivariate Gaussian Discriminant Functions are used for the classification process. Hence $P(x|C_i)$ is assumed to be normally distributed.

Moreover, each feature vector is assumed to follow a multivariate normal distribution and to be statistically independent given a class $C_i$, which results in

$$P(x|C_i) = \frac{1}{\sqrt{2\pi}\sigma_{ik}} e^{[-\frac{(x_i - \mu_{ik})^2}{2\sigma_{ik}^2}}$$

Applying the monotonic logarithm on both sides gives:

$$\log(P(\vec{x}|C_k) = -\sum_{i=1}^{D}[\log(\sqrt{2\pi}\sigma_{ik}) + \frac{(x_i - \mu_{ik})^2}{2\sigma_{ik}^2}] \quad (5)$$

Where D is the Dimension of the feature vectors, and k is the class. $\sigma_{ik}^2$ and $\mu_{ik}$ are the variance and the mean of feature I in class k respectively.

Since the feature vectors are statistically independent, then the non-diagonal elements of the covariance matrix Σ are null. Moreover, each feature vector $x_i$ has the same variance $\sigma_{ik}^2$ [10]. This results in equation (5).

Thus the decision corresponds to the maximum value of the logarithm, described as:

$$\hat{k} = \arg max_{k\in[1,M]} \log(P(\vec{x}|C_k) + \log(P(C_k))$$

Where k is the class index, hence all classes are tested.

2- **Learning**:

This classifier is trained on 36 sets of data and then tested on 16.
The training process is determined by learning the standard deviation and mean of each feature in the training set. This learning is accomplished by taking the partial derivative of the log likelihood L = log(P($X^{(k)}$|$C_k$) with respect to the standard deviation and the mean separately. Where $X^{(k)}$ is the training set containing the feature vectors for each class $C_k$. These partial derivatives are set to zero, in order to find the values of the statistics that maximize the log likelihood, hence maximizing the probability of classifying feature vectors correctly.
Once these parameters are calculated, all that is left to do is to find the prior probability P($C_l$) = $n_l$/N (6). Where $n_l$ is the number of times class k appeared in the data set, and N is the total size of the dataset (Note: dataset means the whole data, i.e. training and testing data combined).
These parameters are then used to test the classifier on unseen data.

3- **Experiments and Results**:

The learned parameters are then passed through the experimental process. All theoretical equations described above are written from scratch using MATLAB and can be found in Appendix B.

The classifier is tested on the test set, by extracting the feature vectors from that set, and calculating the discriminant functions for each class.

The prior probabilities P(S) for “smile” and P(F) for “Frown” are calculated using equation (6), where P(S) = P(F) = 0.5. Then the log likelihood for each class is calculated by passing the test feature vectors and the learned standard deviation and mean from the training set into a MATLAB function that will calculate from scratch equation (5).

These results are then added together with the prior probabilities to calculate the discriminant functions for each class k. Since this is a binary classification case, the following relation was used to assess the accuracy, and thus the error rate of the classifier:

$$\gamma(\mathrm{x}) = \gamma(x)_s - \gamma(x)_f \quad [10]$$

This relation will then deduce that if $\gamma(x) > 0$, classify as “smile”, otherwise classify as “frown”.

This resulting vector containing ones and zeros will be compared to the actual test set to count how many times the classifier chose the right class.

Accuracy of the classifier was found to be: **87.5 %.**

Thus, the error-rate, i.e. the percentage of misclassifying the test feature vectors was found to be: **12.5%.**

In machine learning an accuracy that high is considered to be good. However, this error-rate can be further minimized by adopting different models to this problem. One can also use several models together such as Ensemble Learning in order to further increase this accuracy.

All calculations have been implemented using MATLAB. The log likelihood value has been extracted by a MATLAB function written from scratch. Finding the discriminant functions and testing the results were also implemented in MATLAB. The code can be found attached in Appendix B.

## IV- Conclusion:

To sum up, this exercise used a gaussian discriminant function-based classifier to predict the classes of 17 action units describing specific facial muscle movements. The theory behind gaussian discriminant functions and Naïve bayes classifiers has been depicted clearly. Then used that theory in a practical approach using MATLAB to develop these theoretical functions. A loss of 12.5% was found, which is fairly a good percentage which can be further decreased. However, the aim of this exercise was to demonstrate how Bayes Classifiers combined with gaussian discriminant functions can be trained and further on tested on unseen data. This approach gives an opening to using these classifiers even more in the field of Facial Expression Analysis.

## V- References:

[1] Richmond, V., McCroskey, J., and Payne, S. (1991), Nonverbal behavior in interpersonal relations 526 (Prentice Hall).

[2] D. C. Howell, Statistical methods for psychology, 3rd ed. Boston, MA, US: PWS-Kent Publishing Co, 1992.

[3] Vinciarelli, A., Chatziioannou, P. and Esposito, A., 2015. When the Words are Not Everything: The Use of Laughter, Fillers, Back-Channel, Silence, and Overlapping Speech in Phone Calls. *Frontiers in ICT*, 2.

[4] Provine, R., and Yong, Y. (1991). Laughter: a stereotyped human vocalization. Ethology 89, 115–124. doi:10.1111/j.1439-0310.1991.tb00298.x.

[5] Tian YL., Kanade T., Cohn J.F. (2005) Facial Expression Analysis. In: Handbook of Face Recognition. Springer, New York, NY. https://doi.org/10.1007/0-387-27257-7_12.

[6] Ekman and Friesen, "Measuring Facial Movements", Environmental Psychology and Nonverbal Behavior, 1(1):56-75, 1976.

[7] Tian Y.-I., Kanade T., Cohn J. F. (2001). Recognizing action units for facial expression analysis. IEEE Trans. Pattern Anal. Mach. Intell. 23 97–115. 10.1109/34.908962.

[8] imotions. 2020. *Facial Action Coding System (FACS) - A Visual Guidebook - Imotions*. [online] Available at: <https://imotions.com/blog/facial-action-coding-system/> [Accessed 4 December 2020].

[9] Baltrušaitis, Robinson, and Morency, "Openface: an open source facial behavior analysis toolkit", IEEE Winter Conference on Applications of Computer Vision, 2016.

[10] Vinciarelli, A., CAMASTRA, F., 2016. *MACHINE LEARNING FOR AUDIO, IMAGE AND VIDEO ANALYSIS*.: SPRINGER.

## Appendix A:

## Main code:

```
clc; clear; format long;
laughter_dataset = readtable("laughter-corpus.csv");
%Initialize Number of subjects in Test
speakers = 120;
female_speakers = 63;
male_speakers = 57;
%Extract mean, std, max and min of total dataset
mean_total = mean(laughter_dataset.Duration);
std_total = std(laughter_dataset.Duration);
max_duration = max(laughter_dataset.Duration);
min_duration = min(laughter_dataset.Duration);
%This histogram shows us that the distribution of Duration of calls
over genders and roles
histogram(laughter_dataset(:,[1,3]).Duration)
ylabel("Gender");
xlabel("Duration of laughter");
title("Histogram of pdf");
%Problem 1:
%Extract dataset demonstrating gender only with corresponding Duration
of call.
gender_dataset = laughter_dataset(:,[1,3]);
total_gender = size(gender_dataset);
%Significance level of test number one with 2 observations male/female,
so dof = L - 1 = 2 - 1 = 1, where L = number of observations.
significance_level = 0.05;
degrees_of_freedom = 1;
%This function will calculate the number of times event laughter
happens for females, and the number of times it happens for males
respectively.
gender_count =
splitapply(@height,gender_dataset(:,2),findgroups(gender_dataset(:,1)))
;
disp(['Number of times laughter events happened for female: ',
num2str(gender_count(1))]);
disp(['Number of times laughter events happened for male: ',
num2str(gender_count(2))]);
%This function will calculate the mean for laughter duration for
females, and the mean for laughter duration for males respectively.
gender_mean =
splitapply(@mean,gender_dataset(:,2),findgroups(gender_dataset(:,1)));
disp(['Mean of duration of laughter events for female: ',
num2str(gender_mean(1))]);
disp(['Mean of duration of laughter events for male: ',
num2str(gender_mean(2))]);
```

```
%This function will calculate the variance for laughter duration for
females, and the mean for laughter duration for males respectively.
gender_variance =
splitapply(@var,gender_dataset(:,2),findgroups(gender_dataset(:,1)));
disp(['Variance of duration of laughter events for female: ',
num2str(gender_variance(1))]);
disp(['Variance duration of laughter events for male: ',
num2str(gender_variance(2))]);
%Insert information into a table for better visualization:
Gender = {'Female';'Male'};
table(Gender,gender_count,gender_mean,gender_variance);
%Since we want to reject the Null Hypotheses that "Women laugh the same
as men", we want to test if there is a statisticaly significant
difference between the observed value of female events and the expected
one under the null hypotheses that women and men laugh the same.
Moreover, we are working with discrete values for each events
(Categorical Variables), hence we use the goodness of fit (one-way
classification) Chi square test.
%Extract Duration of calls for Females
female_durations = gender_dataset(gender_dataset{:,1} == "Female",:);
histogram(female_durations.Duration);
ylabel("Female");
xlabel("Duration of laughter");
title("Histogram of pdf");
%Get the observed value for size of laughter events of Females
observed_female = size(female_durations);
%Expected female events, calculated by applying weight to number of
events
expected_female = size(gender_dataset)*(female_speakers/speakers);
%Extract Duraton of calls for Males
male_durations = gender_dataset(gender_dataset{:,1} == "Male",:);
histogram(male_durations.Duration)
ylabel("Male");
xlabel("Duration of laughter");
title("Histogram of pdf");
%Get the observed value for size of laughter events of Male
observed_male = size(male_durations);
%Expected male events, calculated by applying weight to number of
events
expected_male = size(gender_dataset)*(male_speakers/speakers);
%Calculate chi-square value for female events of laughter
X2_f = chi_square(observed_female(1),expected_female(1));
%Calculate chi-square value for male events of laughter
X2_m = chi_square(observed_male(1),expected_male(1));
%Sum chi square for male and female events following the formula of chi
square:
%X2 = ?(O - E)2/E where sum over all observations L = 2 from k = 1.
X2_gender = X2_f + X2_m;
%Since we have one degree of freedom, and alpha of 0.05, we get the X2
of that significance level at the determined degree of freedom.
X2_significance_level =
chi_square_significance_level(significance_level,degrees_of_freedom);
disp(['Chi Squared value for a significance level of ',
num2str(significance_level), ' is : ',
num2str(X2_significance_level)]);
disp(['Chi Squared value for test number one is: ',
num2str(X2_gender)]);
```

```
%Validate Null Hypothesis to reject it or fail to reject it based on
chi squared value
disp(validate_null_hypothesis_chi_test(X2_gender,X2_significance_level,
significance_level));
%Problem 2:
%Extract role data with Duration
role_dataset = laughter_dataset(:,[2,3]);
total_role = size(role_dataset);
%This function will calculate the number of times event laughter
happens for callers, and the number of times it happens for receivers
respectively.
role_count =
splitapply(@height,role_dataset(:,2),findgroups(role_dataset(:,1)));
disp(['Number of times laughter events happened for callers: ',
num2str(role_count(1))]);
disp(['Number of times laughter events happened for receivers: ',
num2str(role_count(2))]);
%This function will calculate the mean for laughter duration for
callers, and the mean for laughter duration for receivers respectively.
role_mean =
splitapply(@mean,role_dataset(:,2),findgroups(role_dataset(:,1)));
disp(['Mean of times laughter events happened for callers: ',
num2str(role_mean(1))]);
disp(['Mean of times laughter events happened for receivers: ',
num2str(role_mean(2))]);
%This function will calculate the variance for laughter duration for
callers, and the mean for laughter duration for receivers respectively.
role_variance =
splitapply(@var,role_dataset(:,2),findgroups(role_dataset(:,1)));
disp(['Variance of times laughter events happened for callers: ',
num2str(role_variance(1))]);
disp(['Variance of times laughter events happened for receivers: ',
num2str(role_variance(2))]);
%Insert information into a table for better visualization:
Role = {'Caller';'Receiver'};
table(Role,role_count,role_mean,role_variance)
%Since we want to reject the Null Hypotheses that "Callers laugh the
same as receivers", we want to test if there is a statisticaly
significant difference between the observed value of callers events and
the expected one under the null hypotheses that callers and receivers
laugh the same. Moreover, we are working with discrete values for each
events (Categorical Variables), hence we use the goodness of fit (one-
way classification) Chi square test.
%Extract Duration of calls for Callers
caller_durations = role_dataset(role_dataset{:,1} == "Caller",:);
histogram(caller_durations.Duration)
ylabel("Caller");
xlabel("Duration of laughter");
title("Histogram of pdf");
Get the observed value for size of laughter events of Callers
observed_caller = size(caller_durations);
%Expected caller events, calculated by applying weight to number of
events
expected_caller = size(role_dataset)*(caller_speakers/speakers);
%Extract Duraton of calls for Receivers
receiver_durations = role_dataset(role_dataset{:,1} == "Receiver",:);
histogram(receiver_durations.Duration)
```

```
ylabel("Receiver");
xlabel("Duration of laughter");
title("Histogram of pdf");
Get the observed value for size of laughter events of Receiver
observed_receiver = size(receiver_durations);
%Expected receiver events, calculated by applying weight to number of
events
expected_receiver = size(role_dataset)*(receiver_speakers/speakers);
%Calculate chi-square value for caller events of laughter
X2_c = chi_square(observed_caller(1),expected_caller(1));
%Calculate chi-square value for receiver events of laughter
X2_r = chi_square(observed_receiver(1),expected_receiver(1));
%Sum chi square for caller and receiver events following the formula of
chi square:
%X2 = ?(O - E)2/E where sum over all observations L = 2 from k = 1.
X2_role = X2_c + X2_r;
%Since we have one degree of freedom, and alpha of 0.05, we get the X2
of that significance level at the determined degree of freedom.
X2_significance_level =
chi_square_significance_level(significance_level,degree_of_freedom);
disp(['Chi Squared value for a significance level of ',
num2str(significance_level), ' is : ',
num2str(X2_significance_level)]);
disp(['Chi Squared value for test number two is: ', num2str(X2_role)]);
%Validate Null Hypothesis to reject it or fail to reject it based on
chi squared value
disp(validate_null_hypothesis_chi_test(X2_role,X2_significance_level,si
gnificance_level));
%Problem 3:
%This function will calculate the variance for laughter duration for
females, and the mean for laughter duration for males respectively.
gender_std =
splitapply(@std,gender_dataset(:,2),findgroups(gender_dataset(:,1)));
disp(['Standard deviation of duration of laughter events for female: ',
num2str(gender_std(1))]);
disp(['Standard deviation duration of laughter events for male: ',
num2str(gender_std(2))]);
%Insert information into a table for better visualization:
Gender = {'Female';'Male'};
table(Gender,gender_count,gender_mean,gender_std)
%Since we want to reject the Null Hypotheses that "Women's laugh are of
the same duration of men's laugh events", we will use the Student's t-
test to check if there is a statistically significant difference
between the means of two samples men and women. Moreover, since we are
dealing with continuous values (Duration), we choose the t-test. A two-
tailed t-test has been chosen because we care about the direction of
the difference (or comparison).
%Calculate sample variances s1 and s1
s1 = gender_std(1)/sqrt(gender_count(1)); %Sample STD for Female
s2 = gender_std(2)/sqrt(gender_count(2)); %Sample STD for Male
%Get sample means x1 and x2
x1 = gender_mean(1); %Sample mean for Female
x2 = gender_mean(2); %Sample mean for Male
%Calculate t statistic
t = t_test(x1,x2,s1,s2);
%The degrees of freedom of this test is going ot be n1 - 1 + n2 - 1
degrees_of_freedom = (gender_count(1)) - 1 + (gender_count(2) - 1);
```

```
%Get t statistic at critical value of 0.025
t_critical_value =
t_test_significance_level(degrees_of_freedom,significance_level/2);
%Validate Null Hypothesis to reject it or fail to reject it based on t
statistic, if it exceeds  the t at the critical value, then we will
reject the null hypothesis. Otherwise we fail to reject it.
validate_null_hypothesis_t_test(t,t_critical_value,significance_level)
%Extract p value using calculated t statistic and degrees of freedom to
further confirm validation of null hypothesis
p = 1-tcdf(t,degrees_of_freedom);
%Since p < 0.025, we can further demonstrate that the rejection of the
null hypothesis is the right result.
%Problem 4:
%This function will calculate the variance for laughter duration for
callers, and the mean for laughter duration for receivers respectively.
role_std =
splitapply(@std,role_dataset(:,2),findgroups(role_dataset(:,1)));
disp(['Standard Deviation of duration of laughter events for callers:
', num2str(role_std(1))]);
disp(['Standard Deviation of duration of laughter events for receivers:
', num2str(role_std(2))]);
%Insert information into a table for better visualization:
Role = {'Caller';'Receiver'};
table(Role,role_count,role_mean,role_std)
%Since we want to reject the Null Hypotheses that "Caller's laugh are
of the same duration of receiver's laugh events", we will use the
Student's t-test to check if there is a statistically significant
difference between the means of two samples callers and receivers.
Moreover, since we are dealing with continuous values (Duration), we
choose the t-test. A two-tailed t-test has been chosen because we care
about the direction of the difference (or comparison).
%Calculate sample variances s1 and s1
s1 = role_std(1)/sqrt(role_count(1)); %Sample STD for Callers
s2 = role_std(2)/sqrt(role_count(2)); %Sample STD for Receivers
%Get sample means x1 and x2
x1 = role_mean(1); %Sample mean for Callers
x2 = role_mean(2); %Sample mean for Receivers
%Calculate t statistic
t = t_test(x1,x2,s1,s2);
%The degrees of freedom of this test is going ot be n1 - 1 + n2 - 1
degrees_of_freedom = (role_count(1) - 1) + (role_count(2) - 1);
%Get t statistic at critical value of 0.025
t_critical_value =
t_test_significance_level(degrees_of_freedom,significance_level/2);
%Validate Null Hypothesis to reject it or fail to reject it based on t
statistic, if it exceeds  the t at the critical value, then we will
reject the null hypothesis. Otherwise we fail to reject it.
validate_null_hypothesis_t_test(t,t_critical_value,significance_level)
%Extract p value using calculated t statistic and degrees of freedom to
further confirm validation of null hypothesis
p = 1-tcdf(t,degrees_of_freedom);
%Since p < 0.025, we can further demonstrate that the rejection of the
null hypothesis is the right result.
```

**Manual Functions:**

```
function [X2] = chi_square(O,E)
%What this function does is to calculate the X^2 value by the
difference of
%the Observed variables in our data and the expected variables
(Expected
%values of the data under the Null Hypotheses) and we caculate the
squared
%of the difference because we care to see if there is a statistically
%significance between them and not "how" far away they are from each
other.
%Hence omitting any negative values.
%We then divide by the expected values so that we keep the squared
%difference relative to the number of observations.
%This test shows us whether there is a statistical difference between
%observed and expected values.
X2 = ((O-E)^2)/E;
end



function[X2_significance_level]=chi_square_significance_level(significa
nce_level,degree_of_freedom)
%This function returns the chi_square of the corresponding significance
%level alpha and the corresponding degree of freedom.
%Generate array containing all significance levels
significance_levels(1,:) = [0.995 0.99 0.975 0.95 0.9 0.1 0.05 0.025
0.01 0.005];
index_alpha = find(significance_levels == significance_level);
%Generate chi_square table for all degrees of freedom.
for df = 1:degree_of_freedom
chi_square_table(df,:) = chi2inv(1-[0.995 0.99 0.975 0.95 0.9 0.1 0.05
0.025 0.01 0.005],df);
end
X2_significance_level =
chi_square_table(degree_of_freedom,index_alpha);
end

function [result_test] =
validate_null_hypothesis_chi_test(X2_calculated,X2_significance_level,a
lpha)
%This function will compare the chi_square value we got from our
%calculations based on observed and expected values of both female and
%male, and the chi square got from the corresponding significance level
got
%from the table with respective degrees of freedom.
if X2_calculated >= X2_significance_level
result_test = ['Reject Null Hypothesis since Chi Square calculated '
...
'(', num2str(X2_calculated), ') is greater than Chi Square of ' ...
'significance level (', num2str(X2_significance_level), ') with ' ...
num2str((1-alpha)*100), '% confidence level.'];
else
result_test = ['Failed to reject Null Hypothesis since Chi Square
calculated ' ...
'(', num2str(X2_calculated), ') is less than Chi Square of ' ...
```

```
'significance level (', num2str(X2_significance_level), ')'];
end
end

function [t] = t_test(x1,x2,s1,s2)
%T_TEST This function will calculate the t statistic: t = (x1 − x2) /(?
/ ?n1 + ? / ?n2)
%where x1 is the mean of sample 1, x2 is the mean of sample 2, n1 is
the
%size of sample 1, n2 is the size of sample 2, ? / ?n1 = s1 is the
sampled
%variance of sample 1, and ? / ?n2 = s2 is the sampled variance of
sample
%2. The t-test will check if there is a statistically significant
difference
%between the two sample means x1 and x2.
t = (x1 - x2)/sqrt((s1^2 + s2^2));
end

function [t_significance_level] =
t_test_significance_level(degree_of_freedom,significance_level)
%This function returns the t statistic of the corresponding
significance
%level alpha and the corresponding degree of freedom.
%Generate array containing all significance levels
significance_levels(1,:) = [0.995 0.99 0.975 0.95 0.9 0.1 0.05 0.025
0.01 0.005];
index_alpha = significance_levels == significance_level;
%Generate chi_square table for all degrees of freedom.
for df = 1:degree_of_freedom
t_table(df,:) = tinv(1-[0.995 0.99 0.975 0.95 0.9 0.1 0.05 0.025 0.01
0.005],df);
end
t_significance_level = t_table(degree_of_freedom,index_alpha);
end

function [result_test] =
validate_null_hypothesis_t_test(t_calculated,t_critical_value,alpha)
%This function will compare the t value we got from our
%calculations based on the mean of samples from Female duration and
Male duration and the t got from
%the corresponding significance level got from the table with
respective degrees of freedom.
if t_calculated >= t_critical_value
result_test = ['Reject Null Hypothesis since t value calculated ' ...
'(', num2str(t_calculated), ') is greater than t value of ' ...
'significance level (', num2str(t_critical_value), ') with ' ...
num2str((1-alpha)*100), '% confidence level.'];
else
result_test = ['Failed to reject Null Hypothesis since t value
calculated ' ...
'(', num2str(X2_calculated), ') is less than t value of ' ...
'significance level (', num2str(X2_significance_level), ')'];
end
end
```

**Appendix B**:

**Main Code**:

```
clc; clear; format short
%This work will implement a classifier based on Gaussian Discriminant
Functions.
%The data that this classifier will be executed on include 52 feature
vectors extracted from 52 face images, split into 36 feature vectors
for training, and 16 feature vectors for testing the classifier.
%Each eature vector includes 17 components that ccount for the
activation level of 17 Action Units.
%The classifier has two classification classes "Smile" or "Frown".
%Start by Extracting the training part of the data
training_data = extract_data("training-part-2.csv");
%Extract features from data table and convert format into matrix
features_table = training_data(:,[1:end-1]);
features_data = table2array(features_table);
%Extract features i for each class k and turn into matrices
features_smile = table2array(training_data(training_data{:,end} ==
"smile",(1:end-1)));
features_frown = table2array(training_data(training_data{:,end} ==
"frown",(1:end-1)));
%Extract mean for each feature i from class k
features_mean_smile = mean(features_smile,1);
features_mean_frown = mean(features_frown,1);
%Extract standard deviation for each feature i from class k
features_std_smile = std(features_smile,1);
features_std_frown = std(features_frown,1);
%Extract testing data to apply classifier by using learned standard
deviation and mean of feature vectors in training data
test_data = extract_data("test-part-2.csv");
%Extract features from tes data table and convert format into matrix
features_table_test = test_data(:,[1:end-1]);
features_data_test = table2array(features_table_test);
%Extract features i for each class k and turn into matrices
features_smile_test = table2array(test_data(test_data{:,end} ==
"smile",(1:end-1)));
features_frown_test = table2array(test_data(test_data{:,end} ==
"frown",(1:end-1)));
%Extract classes from test data table and convert format into matrix
classes_table_test = test_data(:,[end]);
classes_data_test = table2array(classes_table_test);
%Calculate prior probabilities for each class using whole dataset
p_smile = (height(features_smile_test) +
height(features_smile))/(height(features_data_test) +
height(features_data));
p_frown = (height(features_frown_test) +
height(features_frown))/(height(features_data_test) +
height(features_data));
%Calculate Log Likelihood for testing data for each class using learned
standard deviation and learned mean from training data.
log_likelihood_smile =
(gaussian_discriminant_function(features_data_test,features_mean_smile,
features_std_smile));
```

```
log_likelihood_frown =
(gaussian_discriminant_function(features_data_test,features_mean_frown,
features_std_frown));
%Calculate the gaussian disciminant function for each class
gamma_smile = log_likelihood_smile + log(p_smile);
gamma_frown = log_likelihood_frown + log(p_frown);
%Calculate difference between gaussian discriminant to get this rule :
?(x)  = ?1(x) - ?2(x).
%if ?(x) > 0 then classify as smile, otherwise classify as frown.
(Binary classification case)
gamma_decision = gamma_smile - gamma_frown;
%Determine how many misclassifications did the gaussian model perform
by applying rule determined above
classes_predicted = gamma_decision > 0;
classes_predicted = classes_predicted + 1;
classes_test = grp2idx(classes_data_test);
loss = (1 - sum(classes_predicted ==
classes_test)/height(features_data_test))*100;
```

**<u>Manual Functions</u>**:

```
function [L] = gaussian_discriminant_function(x,mean,std)
%This function will calculate the log likelihood based on learned
standard deviation and mean from the training data and
%apply them on the feature vectors of the testing data
L = -sum(log(sqrt(2*pi)*std) + ((x - mean).^2)/(2*std.^2),2);
end

function [data] = extract_data(csv_file)
%EXTRACT_DATA This function will extract a dataset in MATLAB
%table format
data = readtable(csv_file);
end
```