# VISVESVARAYA TECHNOLOGICAL UNIVERSITY

JNANA SANGAMA, BELAGAVI – 590 018

**An Internship Project Report**
**on**

## *Depression Prediction using Machine Learning*

Submitted in partial fulfillment of the requirements for the VIII Semester of degree of **Bachelor of Engineering in Information Science and Engineering** of Visvesvaraya Technological University, Belagavi

**by**


**Arya Adesh**
**1RN18CS019**


**Under the Guidance of**


**Dr. H R Shashidhar**
**Associate Professor**
**Department of CSE**


ESTD:2001
*An Institute with a Difference*

## Department of Computer Science and Engineering

## RNS Institute of Technology

**Dr. Vishnuvaradhan Road, Rajarajeshwari Nagar post,**
**Channasandra, Bengaluru-560098**

**2021-2022**

# RNS INSTITUTE OF TECHNOLOGY

**Dr. Vishnuvaradhan Road, Rajarajeshwari Nagar post,**
**Channasandra, Bengaluru - 560098**

## DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

## CERTIFICATE

Certified that the Internship work entitled ***Depression Prediction using Machine Learning*** has been successfully completed by **Arya Adesh (1RN18CS019)** a bonafide student of **RNS Institute of Technology, Bengaluru** in partial fulfillment of the requirements of 8th semester for the award of degree in **Bachelor of Engineering in Information Science and Engineering** of **Visvesvaraya Technological University, Belagavi** during academic year **2021-2022**. The internship report has been approved as it satisfies the academic requirements in respect of internship work for the said degree.

---

| **Dr. H R Shashidhar** | **Dr. Kiran P** | **Dr. M K Venkatesha** |
|---|---|---|
| Internship Guide | Professor and HoD | Principal |
| Associate Professor | Department of CSE | RNSIT |
| Department of CSE | RNSIT | |

**External Viva**

| **Name of the Examiners** | **Signature with Date** |
|---|---|
| **1.** _____ | **1.** _____ |
| **2.** _____ | **2.** _____ |

# DECLARATION

I, **ARYA ADESH [USN: 1RN18CS019]** student of VIII Semester BE, in Computer Science and Engineering, RNS Institute of Technology hereby declare that the Internship work entitled ***Depression Prediction using Machine Learning*** has been carried outby us and submitted in partial fulfillment of the requirements for the *VIII Semester degree of* ***Bachelor of Engineering in Computer Science and Engineering*** *of Visvesvaraya Technological University, Belagavi* during academic year 2021-2022.

Place: Bengaluru

Date:

**ARYA ADESH**

**(1RN18CS019)**

# ABSTRACT

Depression (major depressive disorder) is a common and serious medical illness that negatively affects how you feel, the way you think and how you act. Fortunately, it is also treatable. Depression causes feelings of sadness and/or a loss of interest in activities you once enjoyed. It can lead to a variety of emotional and physical problems and can decrease your ability to function at work and at home.

Anxiety is a feeling of fear, dread, and uneasiness. It might cause you to sweat, feel restless and tense, and have a rapid heartbeat. It can be a normal reaction to stress. For example, you might feel anxious when faced with a difficult problem at work, before taking a test, or before making an important decision. It can help you to cope. The anxiety may give you a boost of energy or help you focus. But for people with anxiety disorders, the fear is not temporary and can be overwhelming.

The DASS-42 is a 42 item self-report scale designed to measure the negative emotional states of depression, anxiety and stress. The principal value of the DASS in a clinical setting is to clarify the locus of emotional disturbance, as part of the broader task of clinical assessment.

As the scales of the DASS have been shown to have high internal consistency and to yield meaningful discriminations in a variety of settings, the scales should meet the needs of both researchers and clinicians who wish to measure current state or change in state over time.

# ACKNOWLEDGMENT

The fulfillment and rapture that go with the fruitful finishing of any assignment would be inadequate without the specifying the people who made it conceivable, whose steady direction and support delegated the endeavors with success.

I would like to profoundly thank **Management** of **RNS Institute of Technology** for providing such a healthy environment to carry out this Project work.

I would like to express my thanks to our Principal **Dr. M K Venkatesha** for his support and inspired me towards the attainment of knowledge.

I wish to place on record my words of gratitude to **Dr. Kiran P.,** Professor and Head of the Department, Computer Science and Engineering, for being the enzyme and master mind behind my Project work.

I would like to express my profound and cordial gratitude to my Guide **Dr. H R Shashidhar**, Associate Professor, Department of Computer Science and Engineering for his valuable guidance in preparing Project report.

I would like to thank all other teaching and non-teaching staff of Computer Science & Engineering who have directly or indirectly helped me to carry out the project work.

And lastly, I would hereby acknowledge and thank my parents who have been a source of inspiration and also instrumental in carrying out this Project work.

**ARYA ADESH** **USN- 1RN18CS019**

# TABLE OF CONTENTS

**CERTIFICATE**

**ABSTRACT** **i**

**ACKNOWLEDGMENT** **ii**

**TABLE OF CONTENTS** **iii**

**ABBREVIATIONS** **iv**

**1 INTRODUCTION** **1**

1.1 ORGANIZATION 1

1.1.1 Company Profile 2

1.1.2 Domain/ Technology 3

1.1.3 Department 3

1.2 PROBLEM STATEMENT 3

1.2.2 Proposed Solutions 3

1.2.2 Problem Formulation 3

**2 REQUIREMENT ANALYSIS, TOOLS & TECHNOLOGIES** **4**

2.1 Hardware & Software Requirements 4

2.2 Functional Requirements 8

2.3 Tools/ Languages/ Platform 6

**3 DESIGN AND IMPLEMENTATION** **7**

3.1 Architecture 7

3.2 Functional modules 10

3.3 Data Visualization 13

3.4 Methods 18

**4 OBSERVATIONS AND RESULTS** **20**

4.1 Training 20

4.2 Testing and Results 21

**5 CONCLUSION AND FUTURE WORK** **24**

5.1 Conclusion 24

5.2 Future enhancement 24

**6 REFERENCES 25**

# ABBREVIATIONS

| | | |
|---|---|---|
| GUI | - | Graphical User Interface |
| RAM | - | Random Access Memory |
| DASS | - | Depression Anxiety Stress Scale |

# CHAPTER 1

# INTRODUCTION

## 1.1 ORGANIZATION

New Age Solutions Technologies (NASTECH) is a Microsoft Partner Network Company and CATC. With the vision to bridge the Academia-Industry Skill gap, we bring to you two of the most accepted concepts that are in demand currently and will huge scope in corporates for years to come.

### 1.1.1 COMPANY PROFILE

NASTECH is formed with the purpose of bridging the gap between Academia and Industry. Nastech is one of the leading Global Certification and Training service providers for technical and management programs for educational institutions. We collaborate with educational institutes to understand their requirements and form a strategy in consultation with all stakeholders to fulfil those by skilling, reskilling and upskilling the students and faculties on new age skills and technologies.

### 1.1.2 DOMAIN/TECHNOLOGY

**Artificial intelligence (AI)** is intelligence demonstrated by machines, as opposed to natural intelligence displayed by animals including humans. Leading AI textbooks define the field as the study of "intelligent agents": any system that perceives its environment and takes actions that maximize its chance of achieving its goals. Some popular accounts use the term "artificial intelligence" to describe machines that mimic "cognitive" functions that humans associate with the human mind, such as "learning" and "problem solving", however, this definition is rejected by major AI researchers.

**Machine learning** (**ML**) is the study of computer algorithms that can improve automatically through experience and by the use of data. It is seen as a part of artificial intelligence. Machine learning algorithms build a model based on sample data, known as "training data", in order to make predictions or decisions without being explicitly programmed to do so. Machine learning algorithms are used in a wide variety of applications, such as in medicine, email filtering, speech recognition, and computer vision, where it is difficult or unfeasible to develop conventional algorithms to perform the needed tasks

### 1.1.3 DEPARTMENT

Stepping into 20th year of academic excellence in the year 2021, the department of Computer Science and Engineering has grown leaps and bounds. Department was established in the year 2001 right from the inception of the institution with the objective to *impart quality technical education in IT field*. Department is also proud to share that it has been accredited by NBA [twice in the year 2011 and 2014 for 3 years]as it follows quality process to groom students to be industry ready and also encourages them to take up higher studies

Vision: Preparing Better Computer Professionals for a Real World

Mission: Imparting solid foundations and applied aspects in both Computer Science Theory and Programming practices. Providing training and encouraging R & D and Consultancy Services in frontier areas of Computer Science and Engineering with a Global outlook. Educating and preparing the graduates, highly sought after, productive, and well-respected for their work culture. Supporting and inducing lifelong learning

## 1.2 PROBLEM STATEMENT

"***Depression Prediction using Machine Learning",*** we make use the DASS42 score to predict the levels and severity of depression, anxiety and stress.

### 1.2.1 Proposed solution

The DASS-42 is a 42 item self-report scale designed to measure the negative emotional states of depression, anxiety and stress. The principal value of the DASS in a clinical setting is to clarify the locus of emotional disturbance, as part of the broader task of clinical assessment.
As the scales of the DASS have been shown to have high internal consistency and to yield meaningful discriminations in a variety of settings, the scales should meet the needs of both researchers and clinicians who wish to measure current state or change in state over time.

If signs of depression have been detected, then it would be desirable to suggest the user to use a self-care chatbot. (We need to think of a way to do this without infringing on the user's privacy, for example, we do not want to send this diagnosis back to the server in its raw form) Perhaps the suggestion of self-care bot can be an automatic feature that is integrated into GBoard upon depression detection, so that the raw data does not need to go back to the centralized server, and does not require revelation of the user's identity.

### 1.2.2 Problem Formulation

In this opportunity we'll go through a very particular topic. We all know the lockdown during the COVID-19 is affecting all of us in different ways, but the most frequents are depression and anxiety which is an expected outcome - the natural responses to confinement are precisely these, and most of the people don't even know it. It's been a hard time, people are afraid of uncertainty, of losing their jobs as many people have already done, the conditions are met for a major emotional imbalance.

Experts recommend to stay away from social media because it accelerates the depression process, and who is depressed already will be even more, however people expressions on it are a key instrument to determine how a population is feeling.

## Overview:

- Users fill the questionnaire that is DASS 42 which is a set of questions that is designed such that it contains all the questions related to depression, anxiety and stress.
- It can also contain additional information about nationality and screen size, religion, gender to gather additional insights about the root causes of these psychological problems.
- The training data is gathered from Kaggle and Open Psychometrics.
- The use of machine algorithms such as Gaussian Naïve Bayes, Random Forest and K-nearest Neighbors helps us predict the levels and severity of depression, anxiety and stress.

# CHAPTER 2

# REQUIREMENT ANALYSIS & TOOLS

Systems analysis is the process of observing systems for troubleshooting or development purposes. It is applied to information technology, where computer-based systems require defined analysis according to their makeup and design.

## 2.1 RESOURCE REQUIREMENTS

### HARDWARE REQUIREMENTS:

- processor : Pentium 4 processor
- Processor Speed : 2.4 GHz
- RAM : 1GB
- Storage Space : 40GB
- Monitor : 1024x768 or 1280x1024

### SOFTWARE REQUIREMENTS:

- IDE : Jupyter Notebook/Kaggle/ Google Collab
- operating system : Windows

## 2.2 Functional Requirements

### Jupyter Notebook

Project Jupyter is a project and community whose goal is to "develop open-source software, open-standards, and services for interactive computing across dozens of programming languages". It was spun off from Ipython in 2014 by Fernando Pérez and Brian Granger. Project Jupyter's name is a reference to the three core programming languages supported by Jupyter, which are Julia, Python and R, and also a homage to Galileo's notebooks recording the discovery of the moons of Jupiter. Project Jupyter has developed and supported the interactive computing products Jupyter Notebook, JupyterHub, and JupyterLab. Jupyter is a NumFOCUS fiscally sponsored project.

Jupyter Notebook can connect to many kernels to allow programming in different languages. A Jupyter kernel is a program responsible for handling various types of requests (code execution, code completions, inspection), and providing a reply. Kernels talk to the other components of Jupyter using ZeroMQ, and thus can be on the same or remote machines. Unlike many other Notebook-like interfaces, in Jupyter, kernels are not aware that they are attached to a specific document, and can be connected to many clients at once. Usually kernels allow execution of only a single language, but there are a couple of exceptions. By default Jupyter Notebook ships with the IPython kernel. As of the 2.3 release (October 2014), there are 49 Jupyter-compatible kernels for many programming languages, including Python, R, Julia and Haskell.

## Python

Python is an interpreted high-level general-purpose programming language. Its design philosophy emphasizes code readability with its use of significant indentation. Its language constructs as well as its object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects.

Python is dynamically-typed and garbage-collected. It supports multiple programming paradigms, including structured (particularly, procedural), object-oriented and functional programming. It is often described as a "batteries included" language due to its comprehensive standard library.

Guido van Rossum began working on Python in the late 1980s, as a successor to the ABC programming language, and first released it in 1991 as Python 0.9.0. Python 2.0 was released in 2000 and introduced new features, such as list comprehensions and a cycle-detecting garbage collection system (in addition to reference counting). Python 3.0 was released in 2008 and was a major revision of the language that is not complete backward-compatible. Python 2 was discontinued with version 2.7.18 in 2020.

Python consistently ranks as one of the most popular programming languages.

## 2.3 Tools, Languages

## Tools used:

- Feature_extraction
- Regular Expression
- Numpy
- Pandas
- matplotlib
- pyplot
- pipeline
- logistic Regression
- confusion matrix

**Language used** **:** Python

**Platform** : Google Colab or Jupyter Notebook

# CHAPTER 3

# DESIGN AND IMPLEMENTATION

The System Design Document describes the system requirements, operating environment, system and subsystem architecture, files and database design, input formats, output layouts, human-machine interfaces, detailed design, processing logic, and external interfaces. Systems design is the process of defining the architecture, modules, interfaces, and data for a system to satisfy specified requirements. Systems design could be seen as the application of systems theory to product development.

## 3.1 ARCHITECTURE

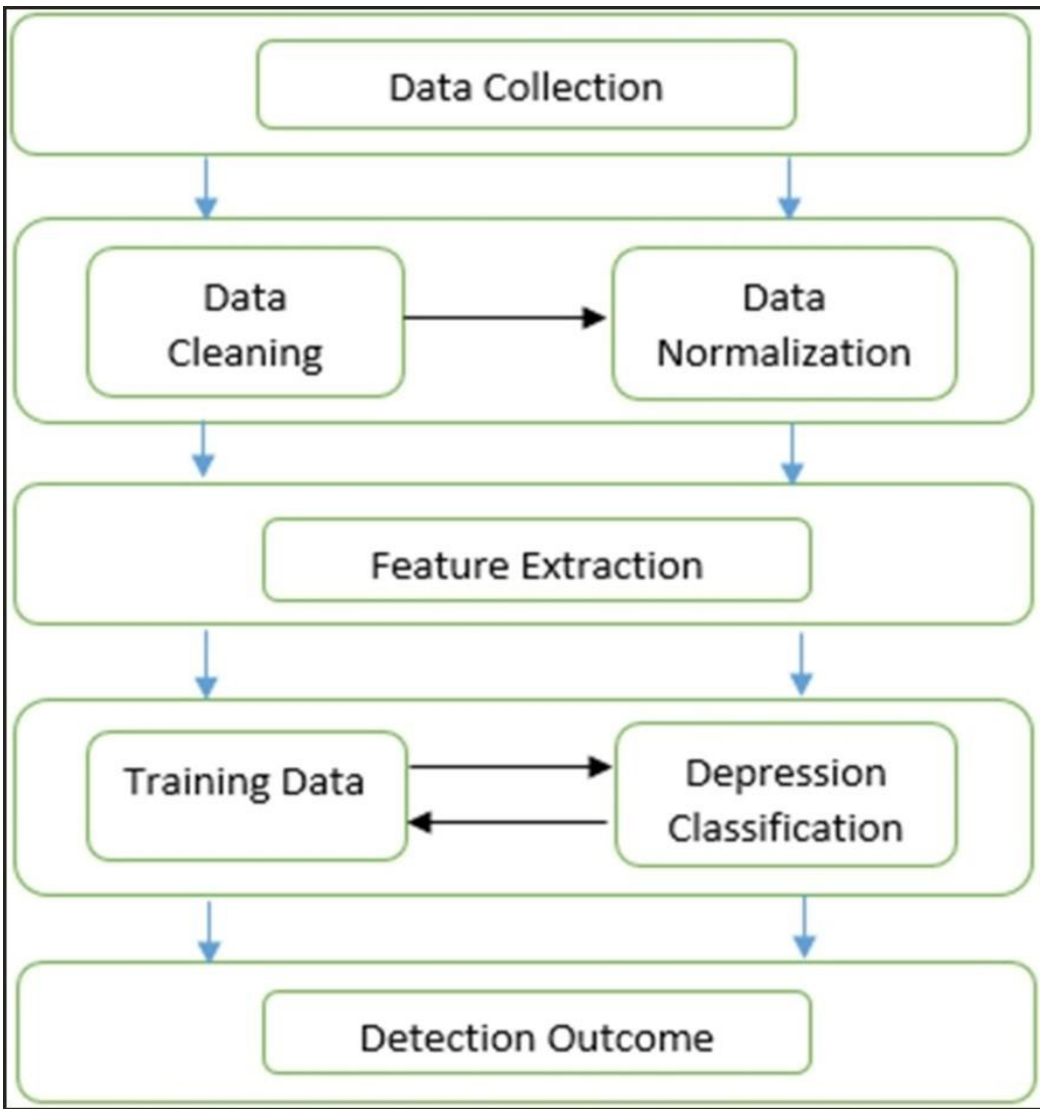



Fig 3.1 Architecture of the project

The Taylor Manifest Anxiety Scale was first developed in 1953 to identify individuals who would be good subjects for studies of stress and other related psychological phenomenon. Since then, it has been used as a measure of anxiety as general personality trait. Anxiety is a complex psychological construct that includes a multiple of different facets related to extensive worrying that may impair normal functioning. The test has been widely studied and used in research however there are some concerns that it does not measure a single trait but instead measures a basket of loosely related ones and so the score is not that meaningful.

The survey was open to anyone and people were motivated to take it to get personalized results. At the end of the test, they also were given the option to complete a short research survey. This dataset comes from those who agreed to complete the research survey and answered yes to the question "Have you given accurate answers and may they be used for research?" at the end.

This data was collected 2017 - 2019.

Gaussian Naive Bayes is a variant of Naive Bayes that follows Gaussian normal distribution and supports continuous data. We have explored the idea behind Gaussian Naive Bayes along with an example.
Before going into it, we shall go through a brief overview of Naive Bayes.

Naive Bayes are a group of supervised machine learning classification algorithms based on the Bayes theorem. It is a simple classification technique, but has high functionality. They find use when the dimensionality of the inputs is high. Complex classification problems can also be implemented by using Naive Bayes Classifier.

Logistic regression is a statistical model that in its basic form uses a logistic function to model a binary dependent variable, although many more complex extensions exist. Mathematically, a binary logistic model has a dependent variable with two possible values, such as pass/fail which is represented by an indicator variable, where the two values are labeled "0" and "1". In the logistic model, the log-odds for the value labeled "1" is a linear combination of one or more independent variables the independent variables can each be a binary or a continuous variable. The corresponding probability of the value labeled "1" can vary between 0 and 1, hence the labeling; the function that converts log-odds to probability is the logistic function, hence the name.

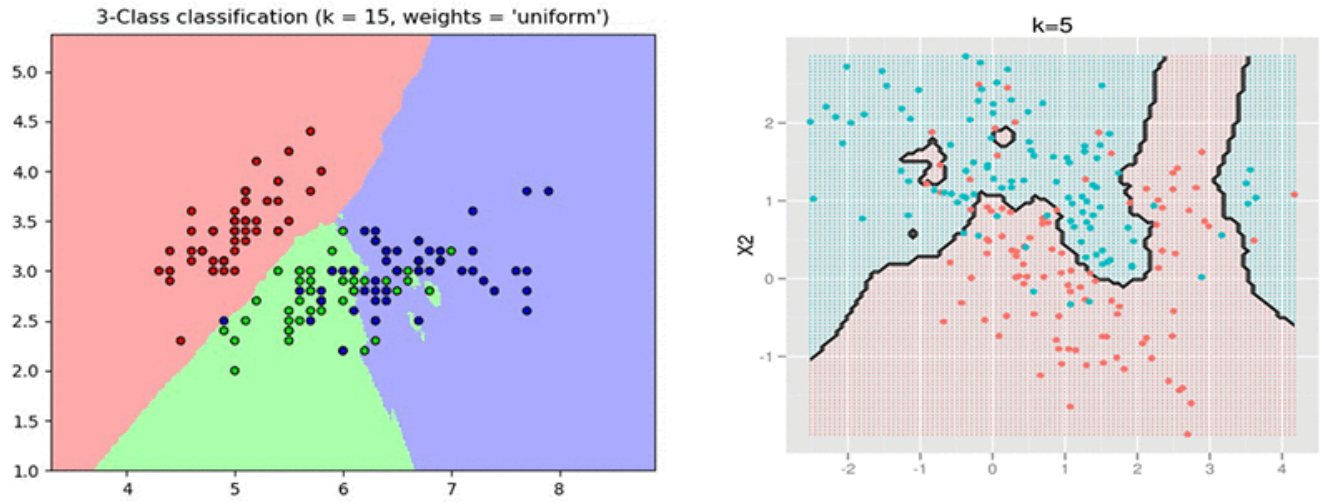



Fig 3.2 KNN In action

K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on Supervised Learning technique. K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories. K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily.

## 3.2 FUNCTIONAL MODULES

```
import numpy as np
import pandas as pd
import seaborn as sn
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import MultinomialNB
import matplotlib.pyplot as plt
from sklearn.metrics import confusion_matrix
from sklearn.metrics import plot_confusion_matrix
from sklearn.metrics import f1_score
from sklearn.ensemble import RandomForestClassifier
from sklearn.neighbors import KNeighborsClassifier
```

Fig 3.2.1 Importing all the Tools

```
df = pd.read_csv('../input/depression-anxiety-stress-scales/DASS_data_21.02.19/data.csv', sep=r'\t', engine='python')
```

```
df.head()
```

| | Q1A | Q1I | Q1E | Q2A | Q2I | Q2E | Q3A | Q3I | Q3E | Q4A | ... | screensize | uniquenetworklocation | hand | religion | orientation | race | vote |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 4 | 28 | 3890 | 4 | 25 | 2122 | 2 | 16 | 1944 | 4 | ... | 1 | 1 | 1 | 12 | 1 | 10 | 2 |
| 1 | 4 | 2 | 8118 | 1 | 36 | 2890 | 2 | 35 | 4777 | 3 | ... | 2 | 1 | 2 | 7 | 0 | 70 | 2 |
| 2 | 3 | 7 | 5784 | 1 | 33 | 4373 | 4 | 41 | 3242 | 1 | ... | 2 | 1 | 1 | 4 | 3 | 60 | 1 |
| 3 | 2 | 23 | 5081 | 3 | 11 | 6837 | 2 | 37 | 5521 | 1 | ... | 2 | 1 | 2 | 4 | 5 | 70 | 2 |
| 4 | 2 | 36 | 3215 | 2 | 13 | 7731 | 3 | 5 | 4156 | 4 | ... | 2 | 2 | 3 | 10 | 1 | 10 | 2 |

Fig 3.2.2 Importing dataset from Drive

```
DASS_keys = {'Depression': [3, 5, 10, 13, 16, 17, 21, 24, 26, 31, 34, 37, 38, 42],
             'Anxiety': [2, 4, 7, 9, 15, 19, 20, 23, 25, 28, 30, 36, 40, 41],
             'Stress': [1, 6, 8, 11, 12, 14, 18, 22, 27, 29, 32, 33, 35, 39]}

DASS_bins = {'Depression': [(0, 10), (10, 14), (14, 21), (21, 28)],
             'Anxiety': [(0, 8), (8, 10), (10, 15), (15, 20)],
             'Stress': [(0, 15), (15, 19), (19, 26), (26, 34)]}
```

```
only_q = df.filter(regex='Q\d{1,2}A')
```

Fig 3.3.3 Cleaning Dataset 1

```
dep = []
for i in DASS_keys["Depression"]:
    dep.append('Q'+str(i)+'A')
stress = []
for i in DASS_keys["Stress"]:
    stress.append('Q'+str(i)+'A')
anx = []
for i in DASS_keys["Anxiety"]:
    anx.append('Q'+str(i)+'A')
```

Fig 3.3.4 Cleaning Dataset 2

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 39775 entries, 0 to 39774
Data columns (total 14 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   Q3A     39775 non-null  int64
 1   Q5A     39775 non-null  int64
 2   Q10A    39775 non-null  int64
 3   Q13A    39775 non-null  int64
 4   Q16A    39775 non-null  int64
 5   Q17A    39775 non-null  int64
 6   Q21A    39775 non-null  int64
 7   Q24A    39775 non-null  int64
 8   Q26A    39775 non-null  int64
 9   Q31A    39775 non-null  int64
 10  Q34A    39775 non-null  int64
 11  Q37A    39775 non-null  int64
 12  Q38A    39775 non-null  int64
 13  Q42A    39775 non-null  int64
dtypes: int64(14)
memory usage: 4.2 MB
```

Fig 3.3.6 Cleaning Dataset 4

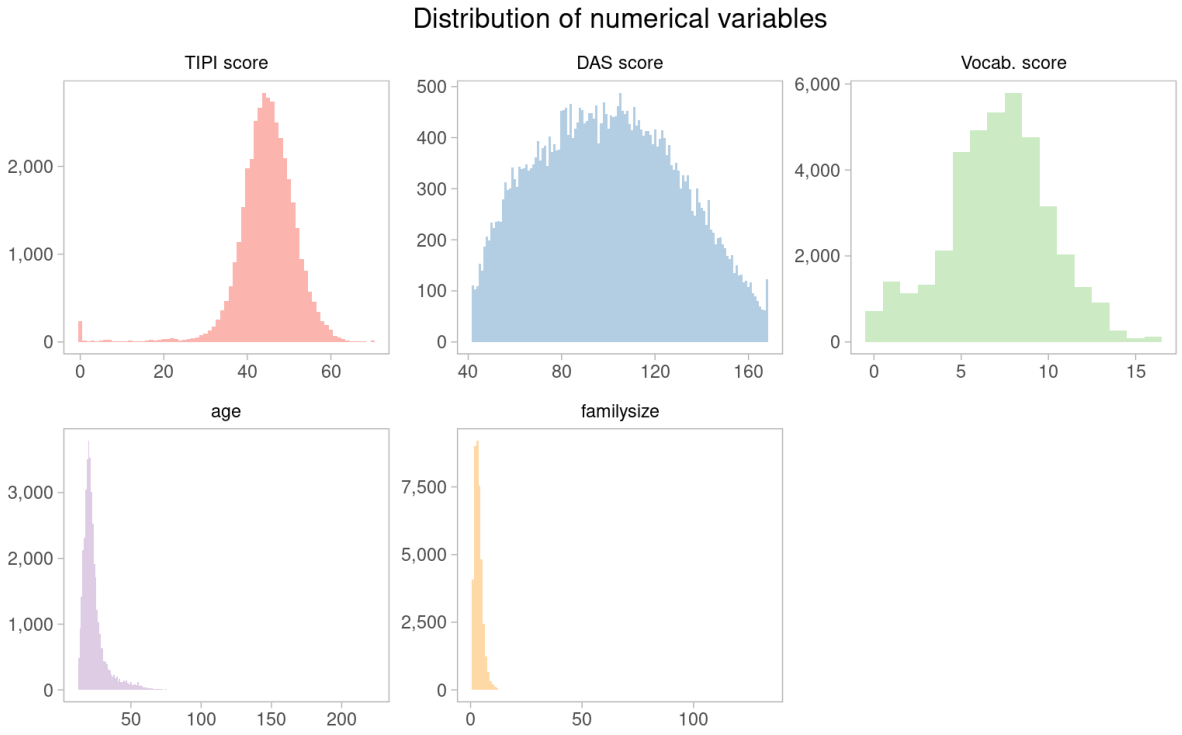
```
train_dep = scores(depression_q)
train_str = scores(stress_q)
train_anx = scores(anxiety_q)
```

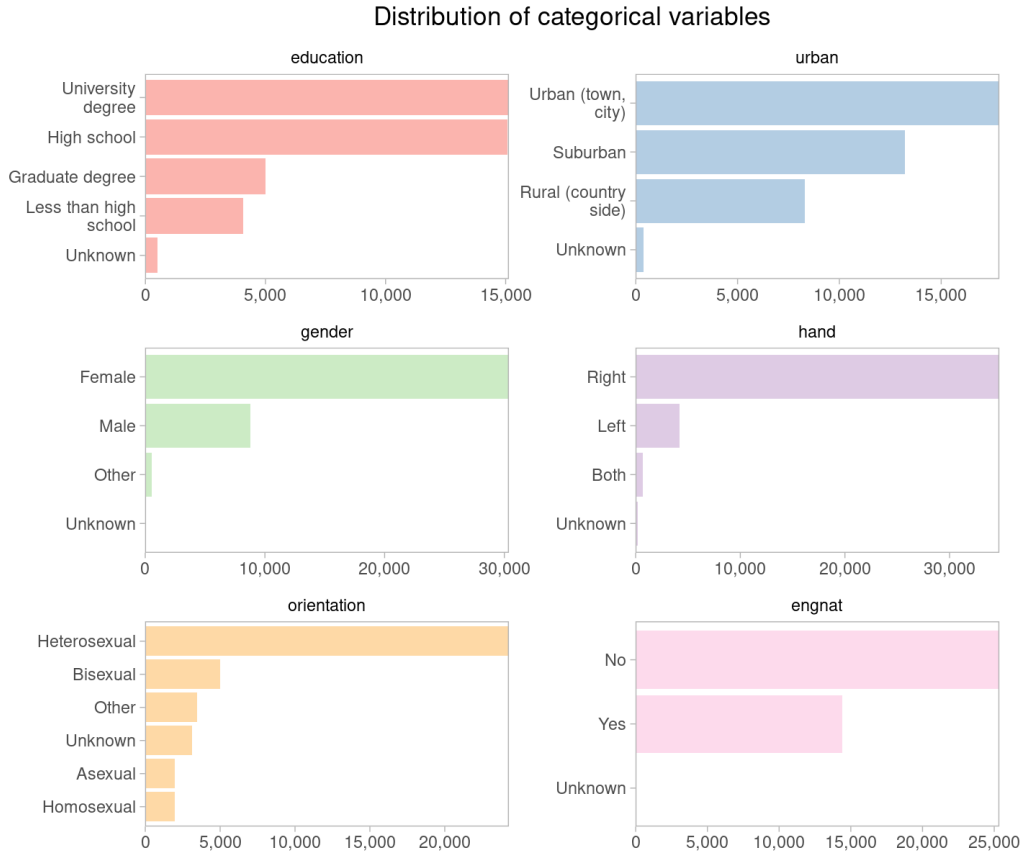
```
train_dep.head()
```

| | Q3A | Q5A | Q10A | Q13A | Q16A | Q17A | Q21A | Q24A | Q26A | Q31A | Q34A | Q37A | Q38A | Q42A | Scores |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 3 | 0 | 3 | 3 | 2 | 0 | 3 | 3 | 3 | 2 | 0 | 1 | 3 | 27 |
| 1 | 1 | 3 | 1 | 3 | 2 | 3 | 1 | 1 | 2 | 1 | 1 | 3 | 1 | 1 | 24 |
| 2 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 0 | 3 | 3 | 3 | 3 | 3 | 39 |
| 3 | 1 | 2 | 2 | 0 | 1 | 2 | 0 | 0 | 1 | 2 | 3 | 1 | 0 | 1 | 16 |
| 4 | 2 | 3 | 2 | 3 | 2 | 3 | 2 | 1 | 3 | 2 | 3 | 2 | 2 | 2 | 32 |

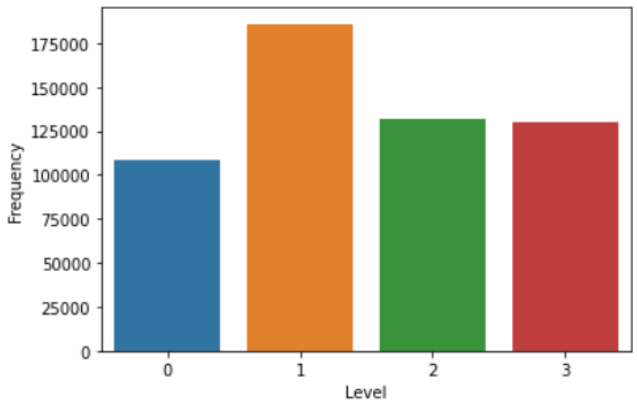
Fig 3.3.7 Cleaning Dataset 5

## 3.3 Data Visualization

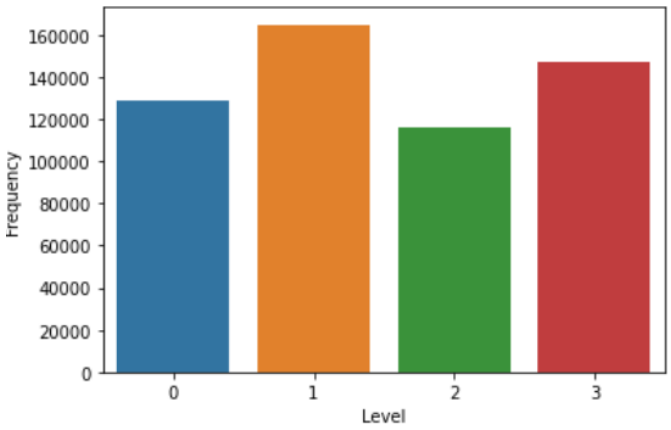


```
draw_freq_plot(stress_q)
```




```
draw_freq_plot(depression_q)
```




```
draw_freq_plot(anxiety_q)
```

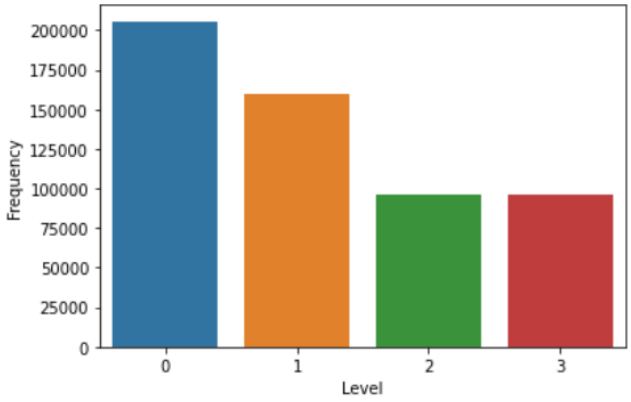



Fig 3.3.8 Data Visualization.

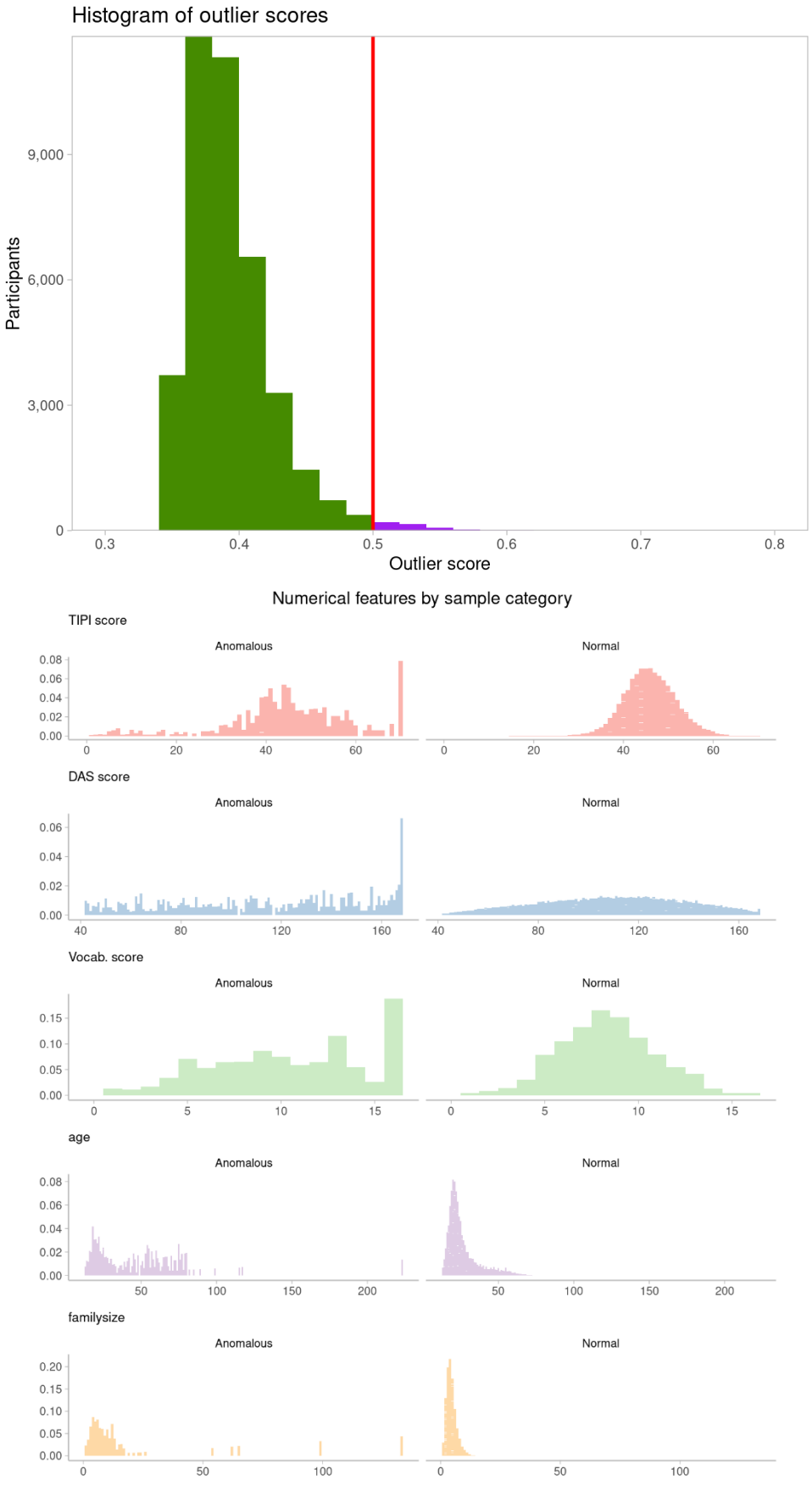
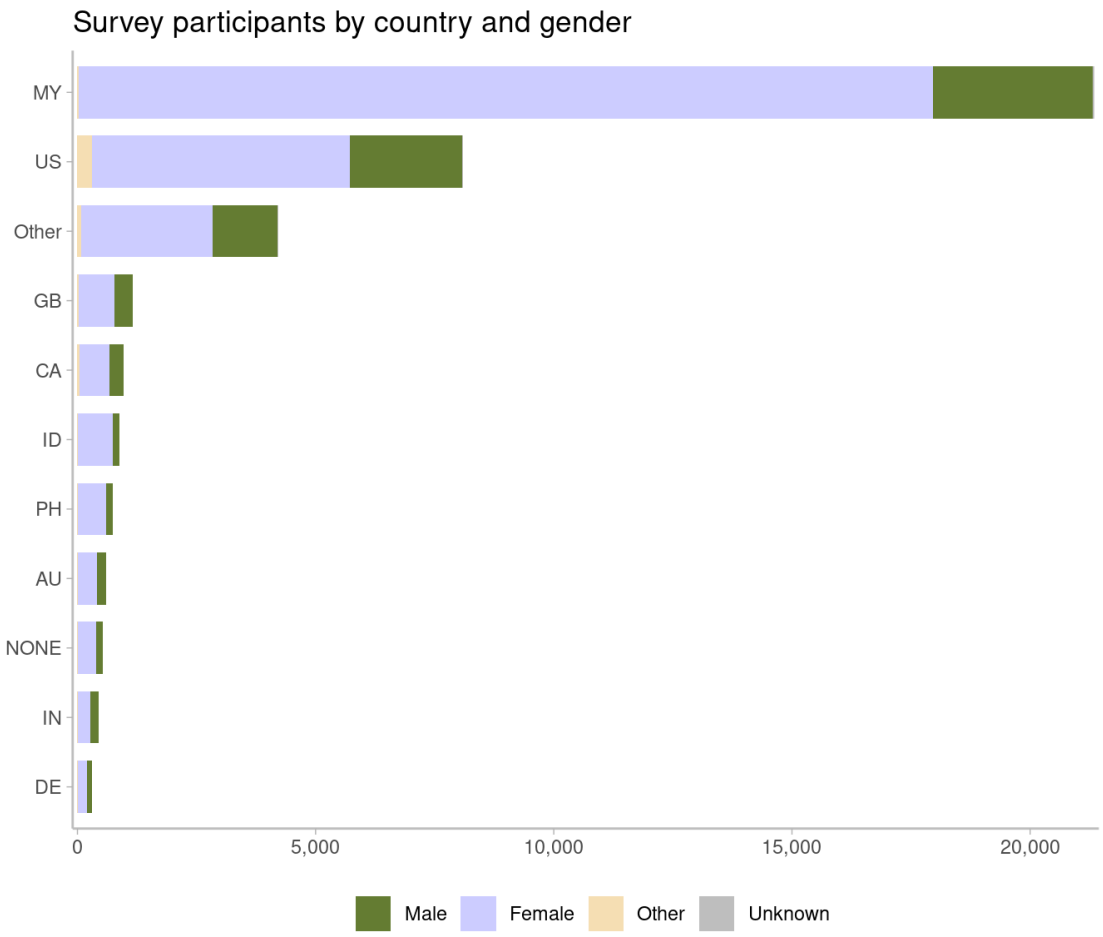
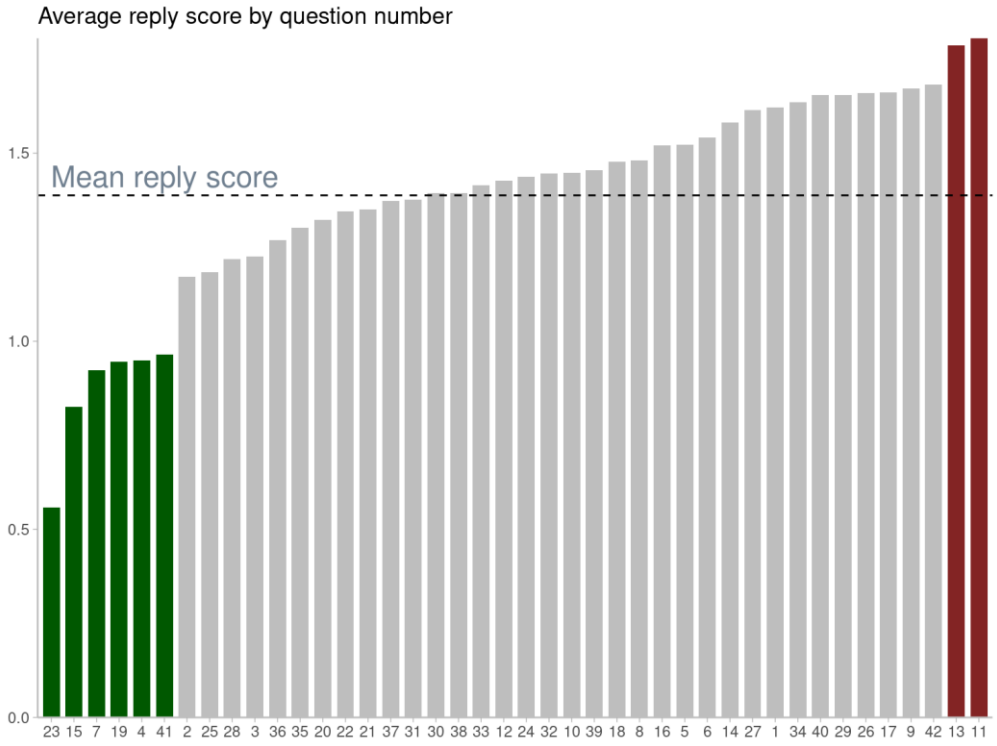
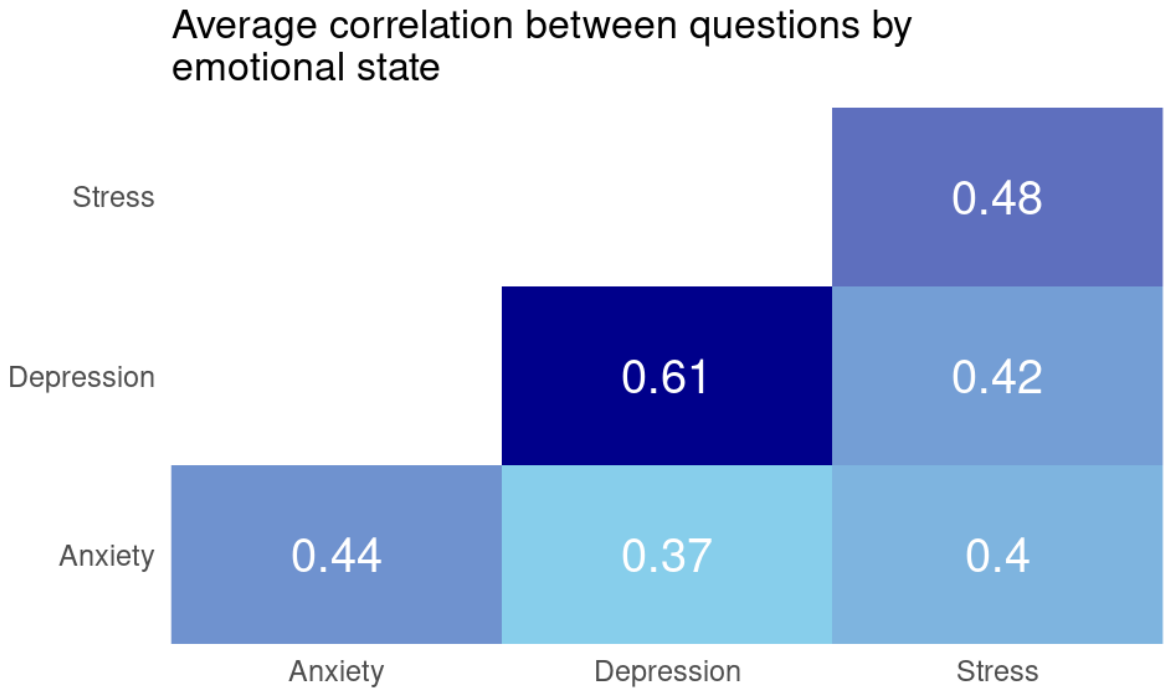



Fig 3.3.9. Data Visualization


Survey participants by country and gender
MY
US
Other
GB
CA
ID
PH
AU
NONE
IN
DE
0
5,000
10,000
15,000
20,000
Male
Female
Other
Unknown


Average reply score by question number
Mean reply score
1.5
1.0
0.5
0.0
23 15 7 19 4 41 2 25 28 3 36 35 20 22 21 37 31 30 38 33 12 24 32 10 39 18 8 16 5 6 14 27 1 34 40 29 26 17 9 42 13 11


Average correlation between questions by emotional state
Stress
Depression
Anxiety
0.48
0.61
0.42
0.44
0.37
0.4
Anxiety
Depression
Stress

## 3.4Methods

```
import seaborn as sn
def draw_freq_plot(df):
    cnt_num = [(df[df == i].sum(axis=1).sum())/(i) for i in range(1,5)]
    ax=sn.barplot(x = [0,1,2,3], y=cnt_num)
    ax.set_xlabel('Level')
    ax.set_ylabel('Frequency')
```

```
def sub(df):
    return df.subtract(1, axis=1)
```

```
def append(df, string):
    conditions = [
    ((df['Scores'] >= DASS_bins[string][0][0])  & (df['Scores'] < DASS_bins[string][0][1])),
    ((df['Scores'] >= DASS_bins[string][1][0])  & (df['Scores'] < DASS_bins[string][1][1])),
    ((df['Scores'] >= DASS_bins[string][2][0])  & (df['Scores'] < DASS_bins[string][2][1])),
    ((df['Scores'] >= DASS_bins[string][3][0])  & (df['Scores'] < DASS_bins[string][3][1])),
    (((df['Scores'] >= DASS_bins[string][3][1])))
    ]
    values = ['Normal','Mild', 'Moderate', 'Severe', 'Extremely Severe']
    df['Category'] = np.select(conditions, values)
    return df

train_dep = append(train_dep, 'Depression')
train_dep.head()
```

```
import matplotlib.pyplot as plt
from sklearn.metrics import confusion_matrix
def plot_confusion_matrix(y_true, y_pred, classes,
                          normalize=False,
                          title=None,
                          cmap=plt.cm.Blues):
    """
    This function prints and plots the confusion matrix.
    Normalization can be applied by setting `normalize=True`.
    """

    title = 'Confusion Matrix'

    # Compute confusion matrix
    cm = confusion_matrix(y_true, y_pred, labels=classes)

    fig, ax = plt.subplots(figsize=(5,5))
    im = ax.imshow(cm, interpolation='nearest', cmap=cmap)
```

```
    ax.set(xticks=np.arange(cm.shape[1]),
           yticks=np.arange(cm.shape[0]),
           # ... and label them with the respective list entries
           xticklabels=classes, yticklabels=classes,
           title=title,
           ylabel='True label',
           xlabel='Predicted label')

    # Rotate the tick labels and set their alignment.
    plt.setp(ax.get_xticklabels(), rotation=0)

    # Loop over data dimensions and create text annotations.
    fmt = '.2f' if normalize else 'd'
    thresh = cm.max() / 2.
    for i in range(cm.shape[0]):
        for j in range(cm.shape[1]):
            ax.text(j, i, format(cm[i, j], fmt),
                    ha="center", va="center",
                    color="white" if cm[i, j] > thresh else "black")
    fig.tight_layout()

    plt.grid(b=None)
```

# CHAPTER 4

# OBSERVATIONS AND RESULTS

## 4.1 TRAINING

One of the key aspects of supervised machine learning is model evaluation and validation. When you evaluate the predictive performance of your model, it's essential that the process be unbiased. Using train_test_split() from the data science library scikit-learn, you can split your dataset into subsets that minimize the potential for bias in your evaluation and validation process.

The training set is applied to train, or fit, your model. For example, you use the training set to find the optimal weights, or coefficients, for linear regression, logistic regression, or neural networks.

```
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import MultinomialNB
cat = train_dep['Category']
train_dep.drop('Category', inplace=True, axis=1)
```

```
Xtrain,Xtest,ytrain,ytest = train_test_split(train_dep, cat, train_size=0.75,random_state=2)
print(Xtrain.shape, ytrain.shape, Xtest.shape, ytest.shape)
```

```
(29831, 15) (29831,) (9944, 15) (9944,)
```

Fig 4.1 Model training

## 4.2 TESTING AND RESULTS:

```
from sklearn.metrics import plot_confusion_matrix
from sklearn.metrics import f1_score
model_d = MultinomialNB(alpha=0.0001, fit_prior = False)
model_d.fit(Xtrain,ytrain)
plot_confusion_matrix(model_d, Xtest, ytest)
predictions = model_d.predict(Xtest)
f1_score(ytest,predictions, average = 'micro')
```

```
0.5294650040225262
```

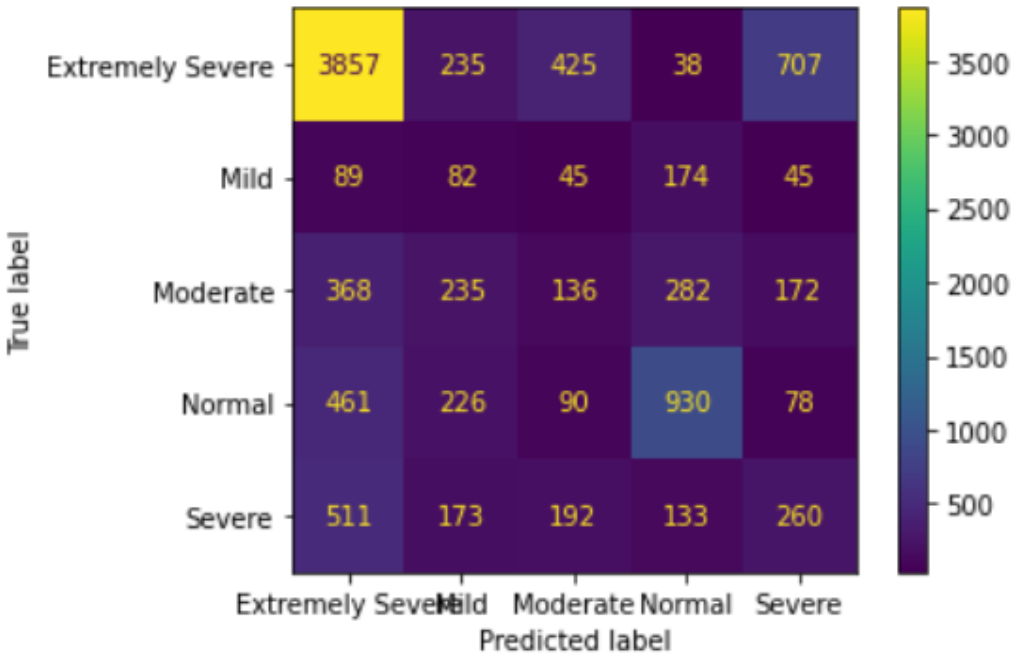



Fig 4.2 Multinomial NB

```
from sklearn.ensemble import RandomForestClassifier
model1 = RandomForestClassifier(random_state=0)
model1.fit(Xtrain,ytrain)
plot_confusion_matrix(model1, Xtest, ytest)
predictions = model1.predict(Xtest)
f1_score(ytest,predictions, average = 'micro')
```

```
1.0
```

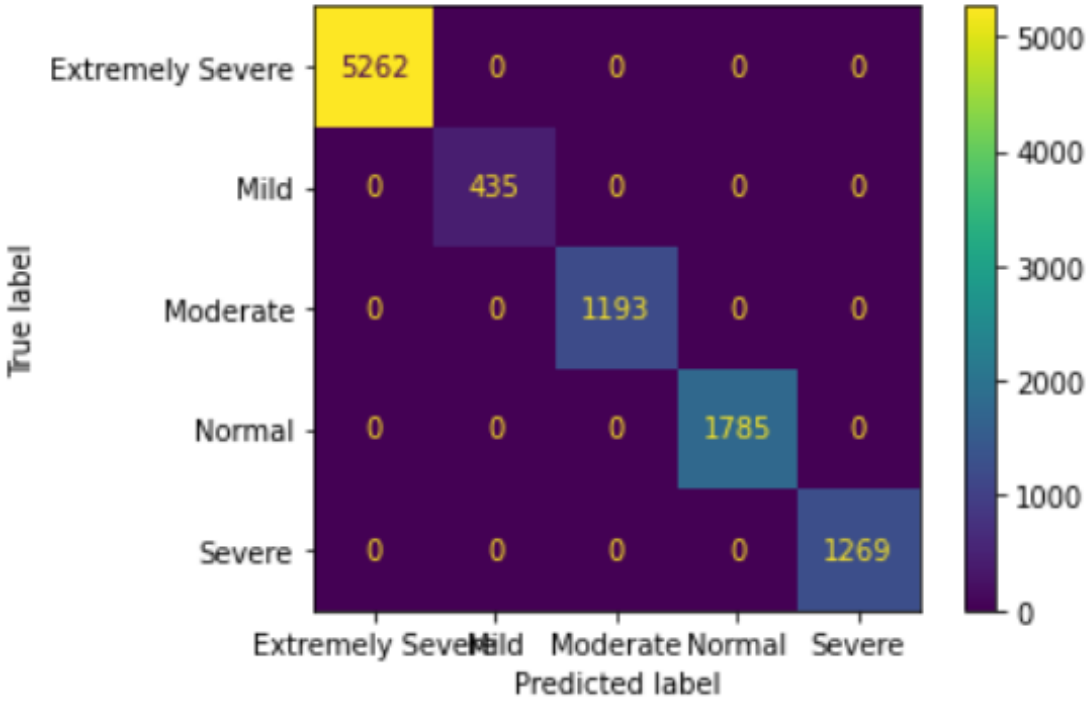



Fig 4.3 Random Forest

```
from sklearn.neighbors import KNeighborsClassifier
model2 = KNeighborsClassifier(n_neighbors=3)
model2.fit(Xtrain,ytrain)
plot_confusion_matrix(model2, Xtest, ytest)
predictions = model2.predict(Xtest)
f1_score(ytest,predictions, average = 'micro')
```

```
0.9725462590506838
```

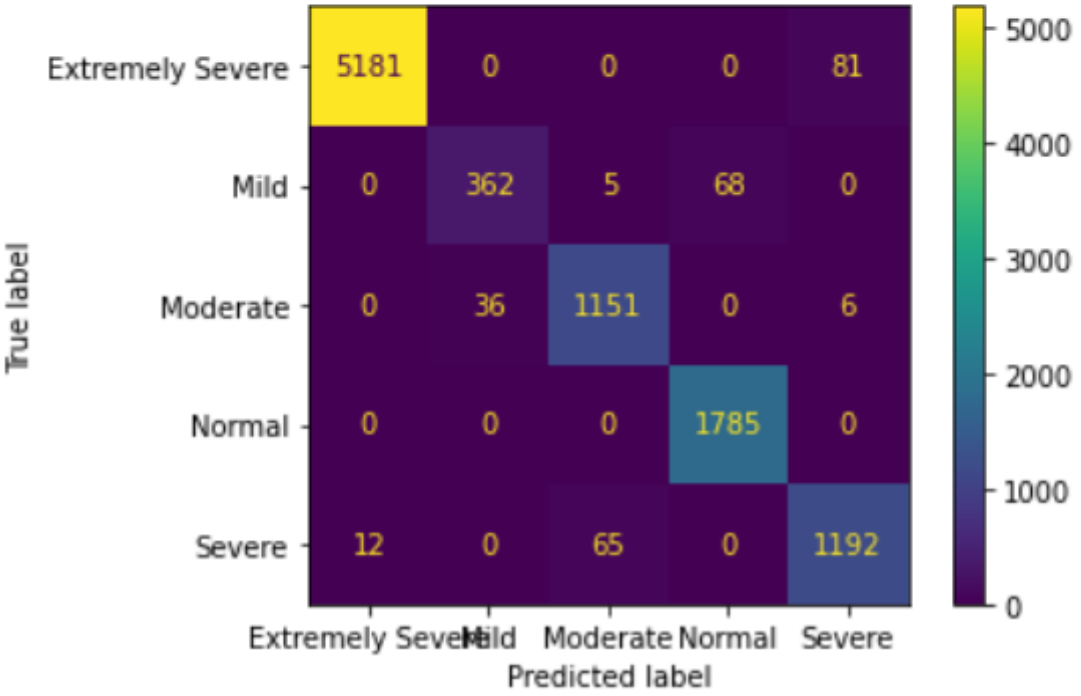



Fig 4.4 K Nearest Neighbors

# CHAPTER 5

# CONCLUSION AND FUTURE ENHANCEMENTS

## 5.1 CONCLUSION

- Using various machine learning algorithms, we were able to classify people into different levels of depression, anxiety and Stress.
- The exploratory data analysis provides insights such as distribution of categorical variables, outliers, survey participants origin country, religion, screen size.
- In conclusion DASS score prediction helped us get better insight into levels of depression, anxiety and stress and factors responsible for them using machine learning and data analysis.

## 5.2 FUTURE ENHANCEMENTS

- Create a website with DASS42 questions and predict the depression, anxiety and stress levels.
- Accessibility for blind people as questions can be read out and they can answer using voice commands.
- Questions can be translated into different languages and this can help a wide array of people.

# CHAPTER 6

# REFERENCES

Links:

Dataset:

1. https://www.kaggle.com/noobiedatascientist/isolation-forest-and-eda/data
2. https://openpsychometrics.org/tests/TMAS/
3. https://www.healthfocuspsychology.com.au/tools/dass-42/
4. https://pdf.sciencedirectassets.com/280203/

Books:

1. Introduction to Machine Learning with Python: A Guide for Data Scientists, by Sarah Guido & Andreas C. Mueller.
2. Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, by Aurelien Geron.
3. Mathematics for Machine Learning**,** by Marc Peter Deisenroth.