UNIVERSITI
MALAYA

WIA1006/WID3006 Machine Learning

Session 2021/2022

Semester 2

Final Report

Title: Depression Indicator

Group Name: Copium

Lecture: K2

Group Member	Matric Number
Leong Jing Wei	U2005251
Aaron Chee Thian Shin	U2102810
Tan Wei Lin	U2102757
Marvin Chin Yi Kai	U2000490
Sizhao Zou	S2104420

Table of Contents

Introduction of Problem

The silent killer that takes lives without warning, punishment, or any sympathy; depression is truly one of the most prominent mental illnesses in the world. Depression is defined as a mental illness including a severe and staunch feeling of sadness. The term depressed is coined in English as a temporary sadness that everyone experiences in their life. Despite that depression is more active among women, it is still one of the most common mental illnesses in the world. It affects anybody, regardless of gender, race, ethnicity, or socioeconomic standing. Regardless of all these facts, surprisingly little is known about depression.

It may even come abruptly, happening in just a few weeks or days. Nervous breakdowns are commonly associated with depression and are often identified due to the confusion and fear that depression brings. Depression heavily influences emotions and one's outlook on life and more than often ends up changing a person's life in a major way. People experiencing depression often feel sad every day and cry very often, making that too another daily routine. Even when participating in activities that used to bring joy, people begin to lose interest and begin secluding themselves from people and things they love. Depression still exists in the teen population and recently has begun increasing in numbers. Reports of depression in teenagers have shown different signs from those of adults.

Suicide remains a major public health problem for the world. The World Health Organization (WHO) estimates that there were over 700,000 deaths from suicide in the world in 2019, with an estimated suicide rate of 9.0 per 100,000 per year. Examining suicide rates and trends are important in shaping national suicide prevention strategies. Suicidal behaviour has been projected to increase globally as a sequela of the anticipated mental health crisis stemming from the COVID-19 pandemic. This mental health crisis is also being referred to as depression as an aftermath of the pandemic. Depression is a common mental disorder. Globally, it is estimated that 5% of adults suffer from the disorder. It is characterised by persistent sadness and a lack of interest or pleasure in previously rewarding or enjoyable activities. Depression and associated mental disorders can have a profound effect on all aspects of life, including performance at school, productivity at work, relationships with family and friends, and the ability to participate in the community. The effects of depression can be long-lasting or recurrent and can dramatically affect a person's ability to function and live a rewarding life. It is shocking that mental illness is rising at an alarming rate and is expected to be the second biggest health problem affecting Malaysians after heart disease by 2020.

There are numerous noticeable symptoms of depression that people have. A lot of people feel hopeless and pessimistic. They also usually feel worthless and helpless. Depression can often coincide with other illnesses or disorders. Such things may precede depression, cause it, and/or be the aftermath of it. Many anxiety disorders accompany depression such as post-traumatic stress disorder, obsessive-compulsive disorder, panic disorder, social phobia, and generalised anxiety disorder. The primary factor contributing to such a horrendous situation is the lackadaisical attitude of Malaysians against mental health issues. This engenders most adults to be unaware of the fact that they might be suffering from depression. There are, in fact, several classifications of depression levels, which are mild and severe, and each level of depression requires a different kind of approach to be tackled. People who suffer from depression do not always experience the same symptoms.

We hope that more people are able to acknowledge the potential of depression that they are having as well as steps that should be taken in order to deal with it. This is why we have decided to develop a machine learning model that can predict the severity of the depression one is facing. By observing some characteristics, this model can predict whether one is having depression or not, as well as determine how critical the depression is. It will also come out with advice on how to deal with it. With that, it can help more people to keep depression at bay. Ahead, we will explain more about how our depression indicator works.

Hypothesis Made for The Problem

By using our depression indicator, users will need to answer a few questions honestly. We believe that the severity of the depression can be predicted, and advice on how to deal with it will also be given based on the severity. Using this model can solve the problem of citizens that are constantly under great stress to avoid mental illness. We expect to see when a user has a lot of motivation to work and feel light-hearted, the result of the depression indicator will show no depression. This model can assist more people to know more about their mental health status.

Project Objectives

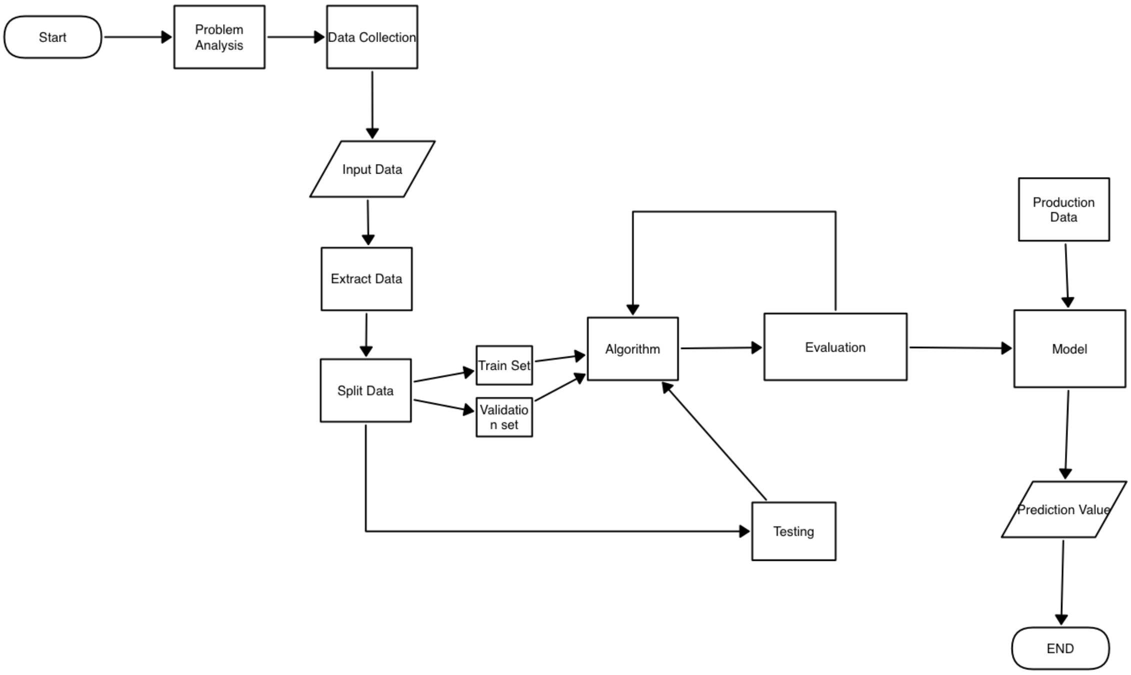
1. **Build a model that can compute the probability of a person suffering from depression, classify the level of depression the person is suffering and recommend a few activities to cope with it.** Throughout this study, we aim to determine the level of depression a person is suffering from by filling in 10 questions. These questions will be used to predict the severity of depression and several activities are recommended for the person to cope with it.

2. **Learn about the technology and common practices for support-vector machines in machine learning.** Our project aims to establish supervised learning models for predicting depression levels by using their responses from 10 survey questions. We also explore different techniques and methods to examine the data.

3. **Develop insights and hypotheses about the nature of learning problems and common approaches for future work.** At the end of this project, we aim to identify the behaviour of different levels of depression to predict depression levels using machine learning methods.

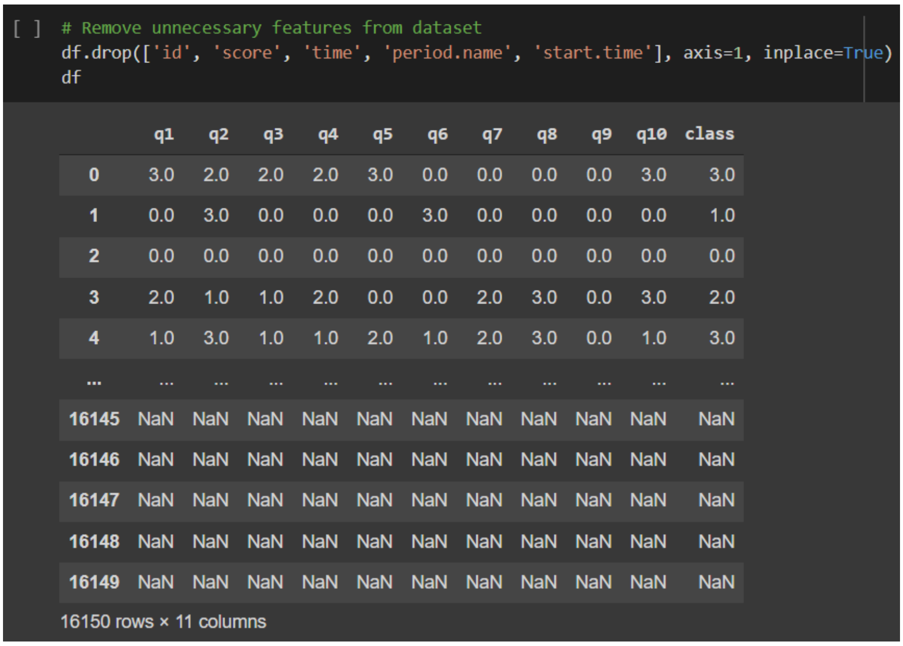
Methodology

There are three main stages in our process: data preprocessing, modelling, and performance analysis. This protocol shows our design to improve the performance of the model.



Data Preprocessing

First, we load the data from the dataset that is uploaded in the drive, then we remove unnecessary features from this dataset to make sure if the dataset contains null values, replace all of them with 0s.

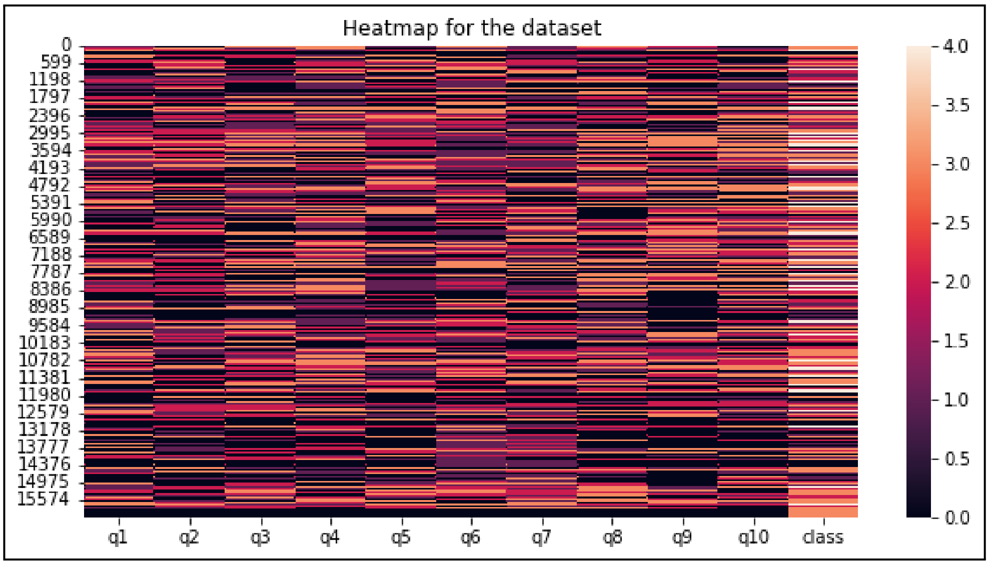


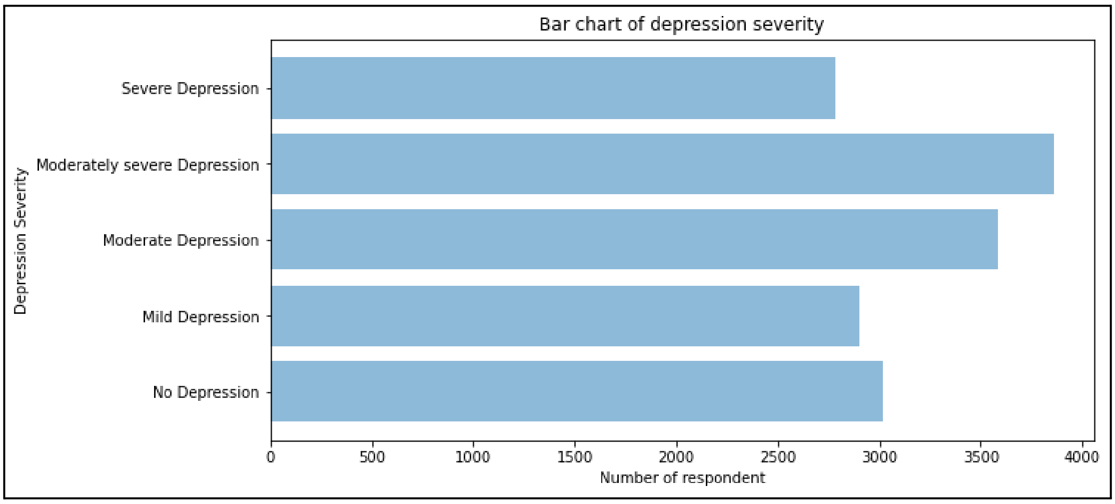
```
[ ]  # Remove unnecessary features from dataset
     df.drop(['id', 'score', 'time', 'period.name', 'start.time'], axis=1, inplace=True)
     df
```

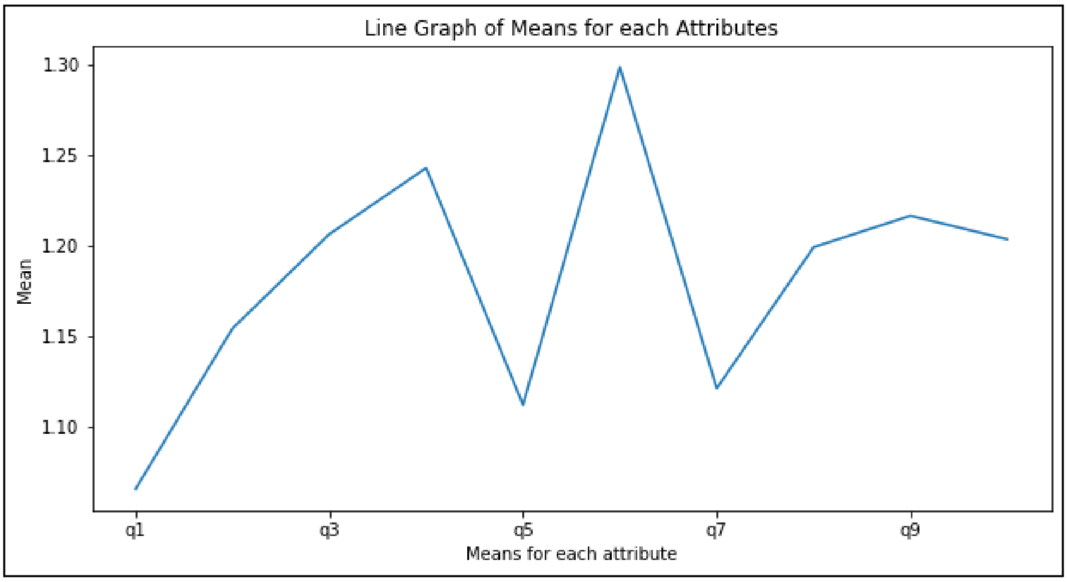
	q1	q2	q3	q4	q5	q6	q7	q8	q9	q10	class
0	3.0	2.0	2.0	2.0	3.0	0.0	0.0	0.0	0.0	3.0	3.0
1	0.0	3.0	0.0	0.0	0.0	3.0	0.0	0.0	0.0	0.0	1.0
2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	2.0	1.0	1.0	2.0	0.0	0.0	2.0	3.0	0.0	3.0	2.0
4	1.0	3.0	1.0	1.0	2.0	1.0	2.0	3.0	0.0	1.0	3.0
...	...	...	...	...	...	...	...	...	...	...	...
16145	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
16146	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
16147	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
16148	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
16149	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

16150 rows × 11 columns

After the data is cleaned, we start to explore and visualise data. First, we used a heatmap to visualise the dataset. Then we used a bar chart of depression severity to compare the severity of depression available in this dataset. Also, we used a line graph of means for each attribute, we allocated the data frame and plotted the mean in the graph.

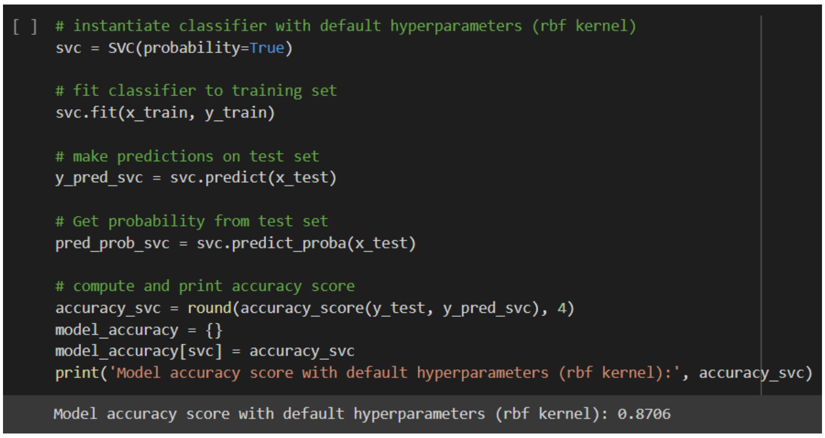




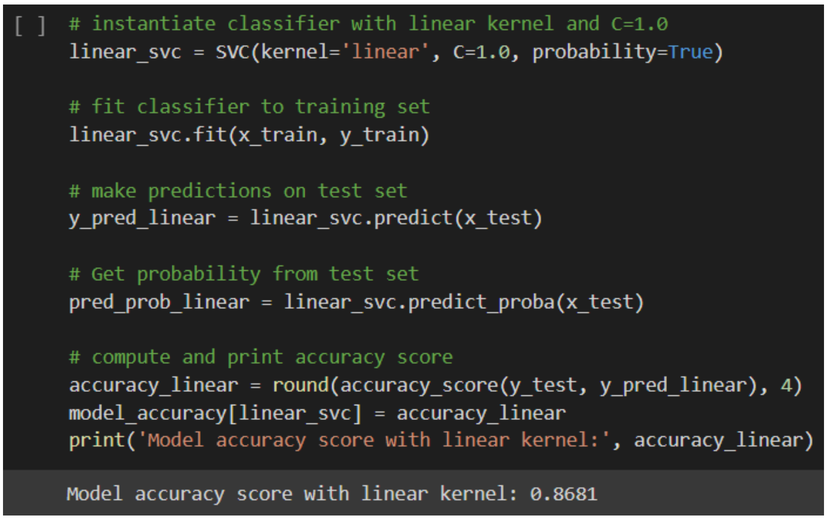


Modelling (Support-Vector machines, SVMs):

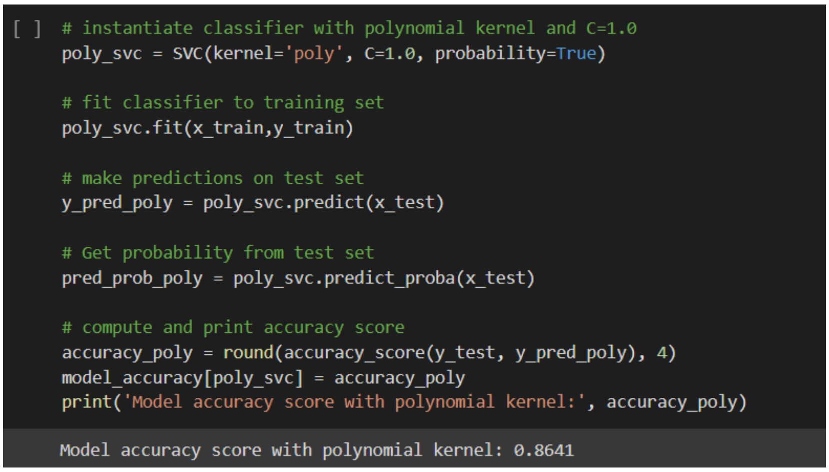
In order to generate our depression indicator, we started feature selection by using Recursive Feature Elimination (RFE). We obtained the list of features selected and split data into 80% training and 20 % testing sets. Support vector machines (SVMs) are a set of supervised learning methods used for classification, regression, and outliers' detection. We used a support vector classifier with several different kernels: RBF kernel, linear kernel, and polynomial kernel. The accuracy score of the RBF kernel is 0.8706, the linear kernel is 0.8681, and the polynomial kernel is 0.8641.



```
[ ] # instantiate classifier with default hyperparameters (rbf kernel)
    svc = SVC(probability=True)

    # fit classifier to training set
    svc.fit(x_train, y_train)

    # make predictions on test set
    y_pred_svc = svc.predict(x_test)

    # Get probability from test set
    pred_prob_svc = svc.predict_proba(x_test)

    # compute and print accuracy score
    accuracy_svc = round(accuracy_score(y_test, y_pred_svc), 4)
    model_accuracy = {}
    model_accuracy[svc] = accuracy_svc
    print('Model accuracy score with default hyperparameters (rbf kernel):', accuracy_svc)

    Model accuracy score with default hyperparameters (rbf kernel): 0.8706
```

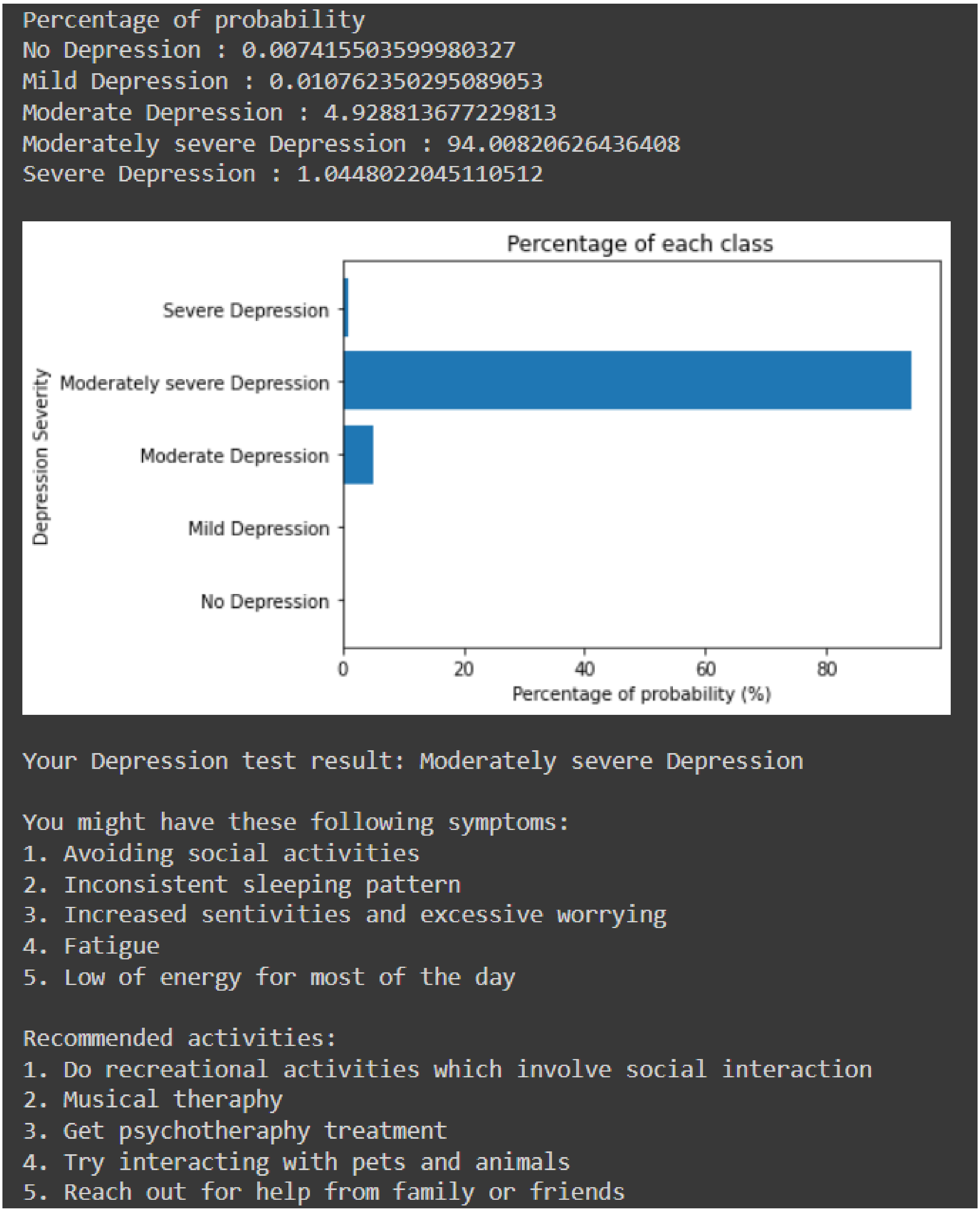


```
[ ] # instantiate classifier with linear kernel and C=1.0
    linear_svc = SVC(kernel='linear', C=1.0, probability=True)

    # fit classifier to training set
    linear_svc.fit(x_train, y_train)

    # make predictions on test set
    y_pred_linear = linear_svc.predict(x_test)

    # Get probability from test set
    pred_prob_linear = linear_svc.predict_proba(x_test)

    # compute and print accuracy score
    accuracy_linear = round(accuracy_score(y_test, y_pred_linear), 4)
    model_accuracy[linear_svc] = accuracy_linear
    print('Model accuracy score with linear kernel:', accuracy_linear)

    Model accuracy score with linear kernel: 0.8681
```



```
[ ] # instantiate classifier with polynomial kernel and C=1.0
    poly_svc = SVC(kernel='poly', C=1.0, probability=True)

    # fit classifier to training set
    poly_svc.fit(x_train,y_train)

    # make predictions on test set
    y_pred_poly = poly_svc.predict(x_test)

    # Get probability from test set
    pred_prob_poly = poly_svc.predict_proba(x_test)

    # compute and print accuracy score
    accuracy_poly = round(accuracy_score(y_test, y_pred_poly), 4)
    model_accuracy[poly_svc] = accuracy_poly
    print('Model accuracy score with polynomial kernel:', accuracy_poly)

    Model accuracy score with polynomial kernel: 0.8641
```

Then, we choose the highest accuracy score of the models and save the model. After that, we load the model and we insert our own inputs for the attributes to get a sample output of the model. The figure below shows the result that we get.

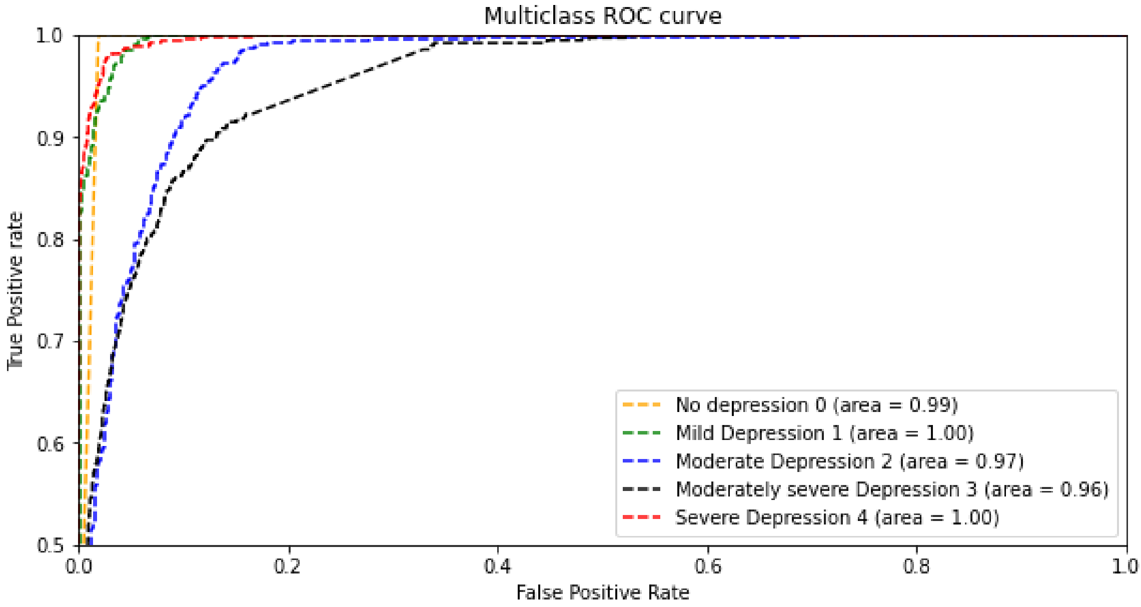


```
Percentage of probability
No Depression : 0.007415503599980327
Mild Depression : 0.010762350295089053
Moderate Depression : 4.928813677229813
Moderately severe Depression : 94.00820626436408
Severe Depression : 1.0448022045110512
```

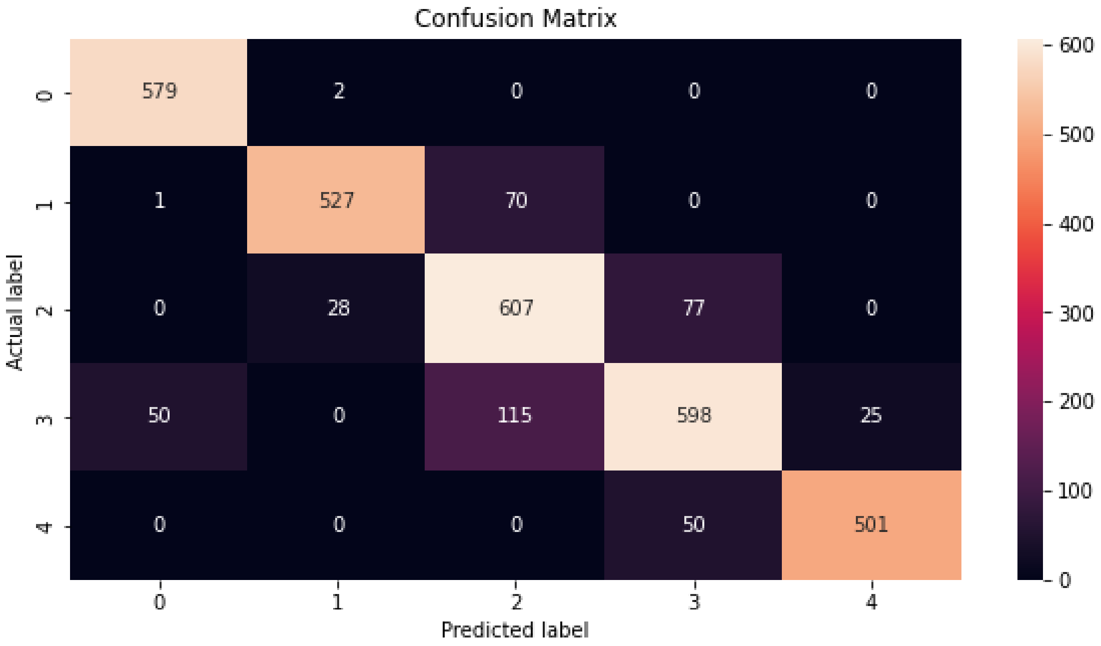
```
Your Depression test result: Moderately severe Depression

You might have these following symptoms:
1. Avoiding social activities
2. Inconsistent sleeping pattern
3. Increased sentivities and excessive worrying
4. Fatigue
5. Low of energy for most of the day

Recommended activities:
1. Do recreational activities which involve social interaction
2. Musical theraphy
3. Get psychotheraphy treatment
4. Try interacting with pets and animals
5. Reach out for help from family or friends
```

Performance Analysis

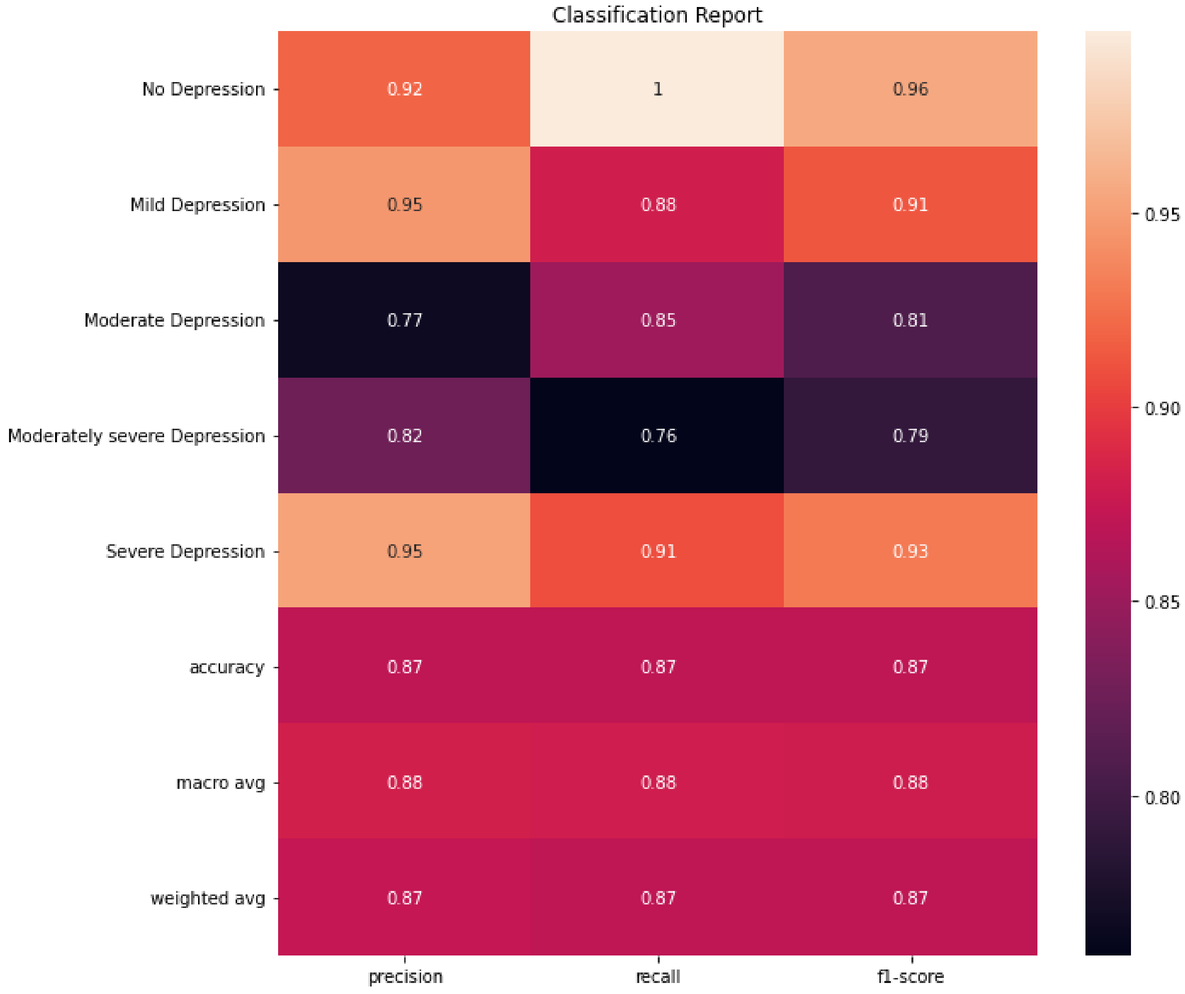
Lastly, after modelling, we built up the analysing codes. In our support vector classifier, we got the highest accuracy by using the RBF kernel. We decided to evaluate the performance of the classifier and analyse the rate of success with the area under the ROC curve(AUC). An excellent model has AUC near the 1 which means it has a good measure of separability. A poor model has an AUC near 0 which means it has the worst measure of separability.



Next, we use a confusion matrix. A confusion matrix is used for evaluating the performance of a classification model, where N is the number of target classes. The matrix compares the actual target values with those predicted by the machine learning model. This gives us a holistic view of how well our classification model is performing and what kinds of errors it is making.



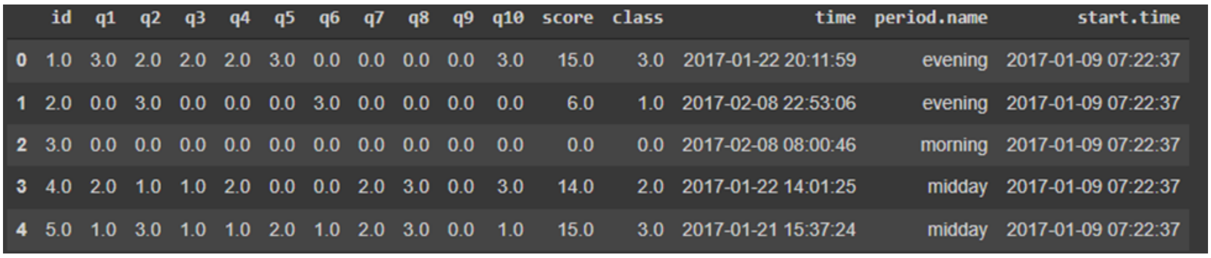
After that, we also created a classification report which consists of precision, recall, F1 score, and support. When everything has been done and the model has been finalized, we started to write the report.



Classification Report

	precision	recall	f1-score
No Depression	0.92	1	0.96
Mild Depression	0.95	0.88	0.91
Moderate Depression	0.77	0.85	0.81
Moderately severe Depression	0.82	0.76	0.79
Severe Depression	0.95	0.91	0.93
accuracy	0.87	0.87	0.87
macro avg	0.88	0.88	0.88
weighted avg	0.87	0.87	0.87

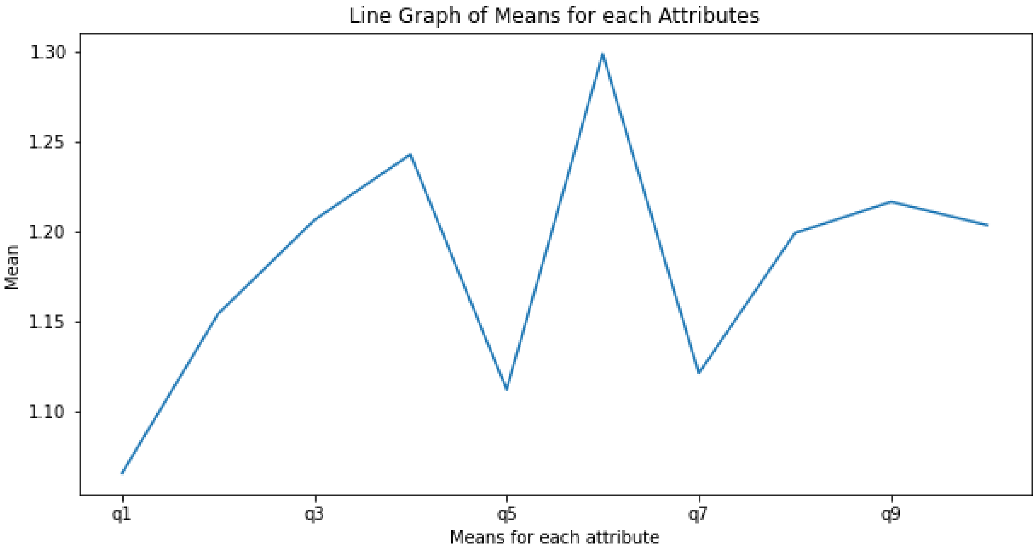
Elaboration on Data & Features Used

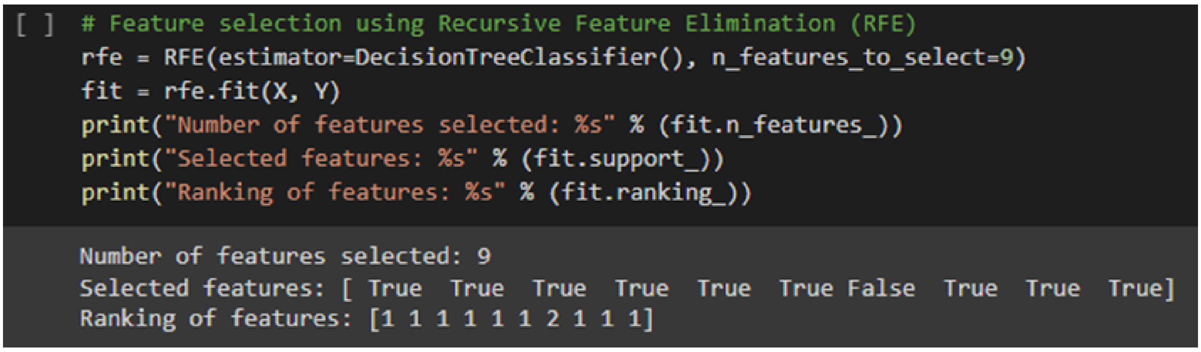


	id	q1	q2	q3	q4	q5	q6	q7	q8	q9	q10	score	class	time	period.name	start.time
0	1.0	3.0	2.0	2.0	2.0	3.0	0.0	0.0	0.0	0.0	3.0	15.0	3.0	2017-01-22 20:11:59	evening	2017-01-09 07:22:37
1	2.0	0.0	3.0	0.0	0.0	0.0	3.0	0.0	0.0	0.0	0.0	6.0	1.0	2017-02-08 22:53:06	evening	2017-01-09 07:22:37
2	3.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	2017-02-08 08:00:46	morning	2017-01-09 07:22:37
3	4.0	2.0	1.0	1.0	2.0	0.0	0.0	2.0	3.0	0.0	3.0	14.0	2.0	2017-01-22 14:01:25	midday	2017-01-09 07:22:37
4	5.0	1.0	3.0	1.0	1.0	2.0	1.0	2.0	3.0	0.0	1.0	15.0	3.0	2017-01-21 15:37:24	midday	2017-01-09 07:22:37

The dataset is the result of survey data collected, which consists of 10 questions, each of which has its own set of answers with its own set of values. The set of questions was designed to gather information about the psychosocial status of the participants. Based on the answers obtained from the survey, it will classify whether or not the respondent is facing depression; if so, it will evaluate the severity of the participants into 5 different sets of classes, each with its own number of classes.

- 0 for “No Depression.”
- 1 for “Mild Depression.”
- 2 for “Moderate Depression.”
- 3 for “moderately severe depression.”
- 4 for “severe depression.”





```
[ ]  # Feature selection using Recursive Feature Elimination (RFE)
     rfe = RFE(estimator=DecisionTreeClassifier(), n_features_to_select=9)
     fit = rfe.fit(X, Y)
     print("Number of features selected: %s" % (fit.n_features_))
     print("Selected features: %s" % (fit.support_))
     print("Ranking of features: %s" % (fit.ranking_))
```

```
Number of features selected: 9
Selected features: [ True  True  True  True  True  True False  True  True  True]
Ranking of features: [1 1 1 1 1 1 2 1 1 1]
```

For our model, we used 9 out of 10 questions available, which were chosen by the feature selection algorithm, Recursive Feature Elimination (RFE), to select the best set of questionnaire data that can be used to train the model to predict and classify the severity of depression faced by participants more accurately. The reason for using RFE is that RFE is given and used in the core of the method, which is wrapped by RFE and can be used to help select features. With a support-vector machine model used as the core of the model, the RFE algorithm helps rank the features by importance, discard the least important features, and refit the model, which will be repeated until a specified number of features remains. since fewer features allow the model to run more efficiently and be more effective in making more accurate classifications.

The cost function that is being minimised for SVM:

$$J = \frac{1}{2}\alpha^{\mathrm{T}} H\alpha - \alpha^{\mathrm{T}} 1$$

With the following constraints:

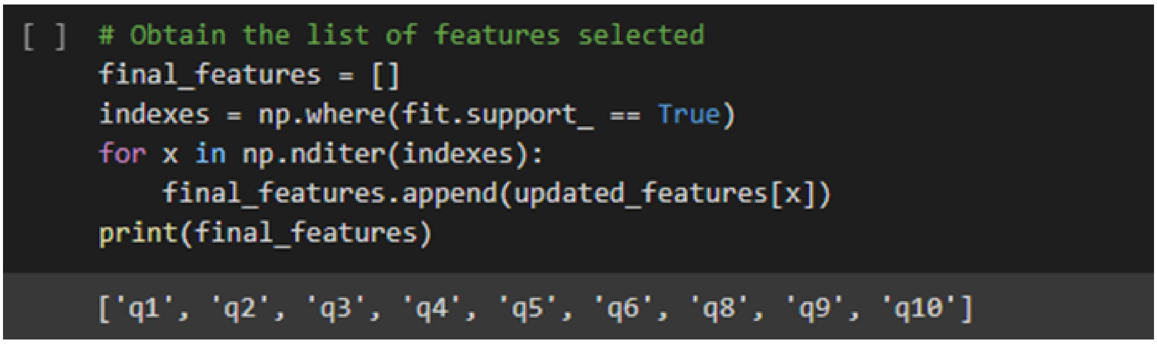
$$0 \leq \alpha_k \leq C, \quad \sum_k \alpha_k y_k = 0$$

Where α is the vector of weights on the training instances learned by the SVM algorithm, is the class value for the training instance, and C is a regularisation parameter.

Matrix H is the kernel matrix for the kernel function K and the set of training instances. In particular, for each pair of training instances such as x_c and x_d , $H = y_c y_d K(x_c, x_d)$.

Finally, for the final selected set of features, the top 9 questions will be the benchmark for the model. We believe that with the use of RFE to select the best features to train the model and support vector machines as the core model, it can increase the accuracy of the model by selecting the 9 features that have a big impact on all the questions inside the questionnaire. It produces a decent result as well, considering by accuracy and recall (sensitivity) number, that the goal of this questionnaire is to predict whether or not the participants face the case of depression and if so, predict the severity of the depression faced by the participants before it goes to any worse extent.

The top 9 features chosen by the RFE algorithm:



```
[ ]  # Obtain the list of features selected
     final_features = []
     indexes = np.where(fit.support_ == True)
     for x in np.nditer(indexes):
         final_features.append(updated_features[x])
     print(final_features)

     ['q1', 'q2', 'q3', 'q4', 'q5', 'q6', 'q8', 'q9', 'q10']
```

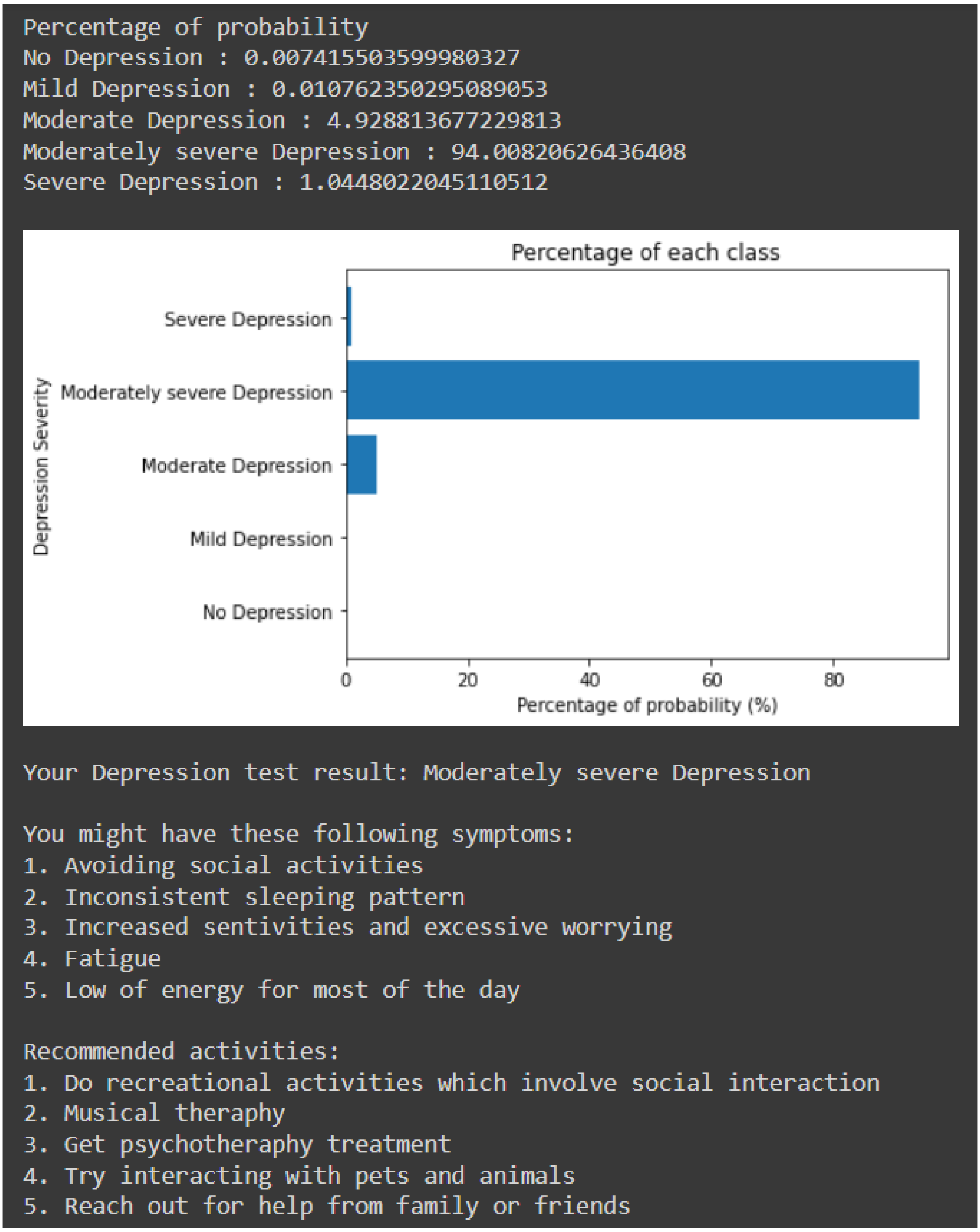
Question 1	Are you have a little interest or pleasure in doing things?
Question 2	Are you feeling down, depressed, or hopeless?
Question 3	Are you having trouble falling or staying asleep, or sleeping too much?
Question 4	Are you feeling tired or having little energy?
Question 5	Are your poor appetite, weight loss or overeating?
Question 6	Are you feeling bad about yourself - or that you are a failure or have let yourself or your family down?
Question 8	Are you feeling slowed down when you are talking to others?
Question 9	Are you that you would be better off dead, or hurting yourself?
Question 10	If you've had any days with the issues above, how difficult have these problems made it for you at work, at home, at school, or with other people?

Based on the list above, the selected feature will have higher importance compared to the rest of the features available. With a smaller number of features selected, the model was able to get a higher score compared to when it did not use RFE for feature selection since it was able to remove some noise from the results. In a nutshell, we believe that RFE will work well for most non-linear types of models and will run relatively quickly. Therefore, since we are using Non-linear support vector machines (SVMs) as our model, RFE should be suitable for our model for feature selection.

Results and Discussions

The picture below is the screenshot of the sample output that our depression indicator generates. It will first display each class's probability percentage (No Depression, Mild Depression, Moderate Depression, Moderately severe Depression, Severe Depression). It generates a horizontal bar chart so that users can visualise the probability distribution better. After that, it shows the depression test result as well as the possible symptoms. Lastly, it recommends some activities to overcome depression.

Screenshot of output:



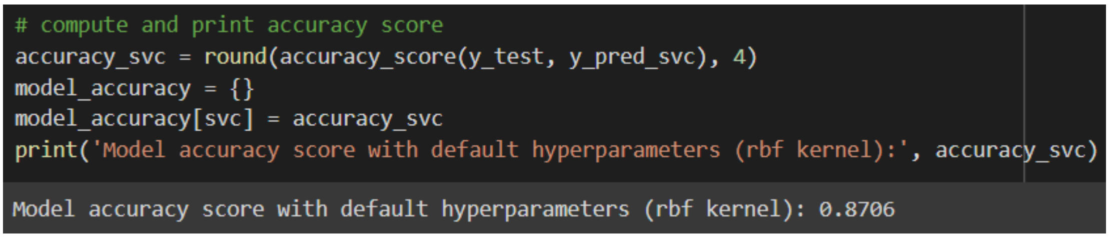
```
Percentage of probability
No Depression : 0.007415503599980327
Mild Depression : 0.010762350295089053
Moderate Depression : 4.928813677229813
Moderately severe Depression : 94.00820626436408
Severe Depression : 1.0448022045110512
```

```
Your Depression test result: Moderately severe Depression

You might have these following symptoms:
1. Avoiding social activities
2. Inconsistent sleeping pattern
3. Increased sentivities and excessive worrying
4. Fatigue
5. Low of energy for most of the day

Recommended activities:
1. Do recreational activities which involve social interaction
2. Musical theraphy
3. Get psychotheraphy treatment
4. Try interacting with pets and animals
5. Reach out for help from family or friends
```

Performance Evaluation

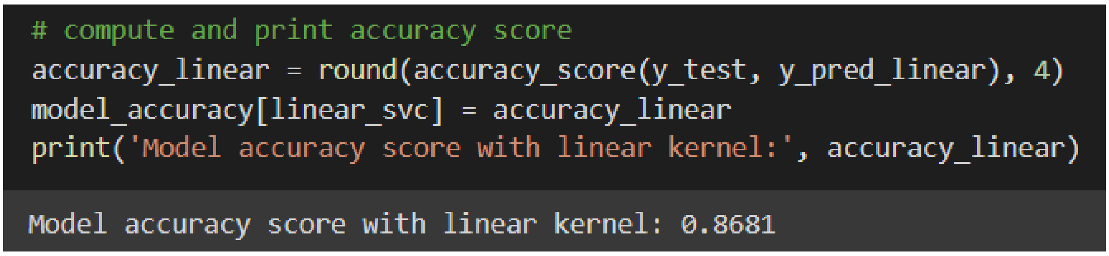
Several methods are implemented in order to improve the accuracy of prediction of our machine learning model on the dataset. In this project, we compare different models and use different hyperparameters so that we can get the best model that suits our dataset. We find the accuracy score of the model, the area under the ROC curve, the classification report which includes precision, recall and F1 score as well as the confusion matrix. Below are the measures we took to conduct the performance analysis:

1) Accuracy score



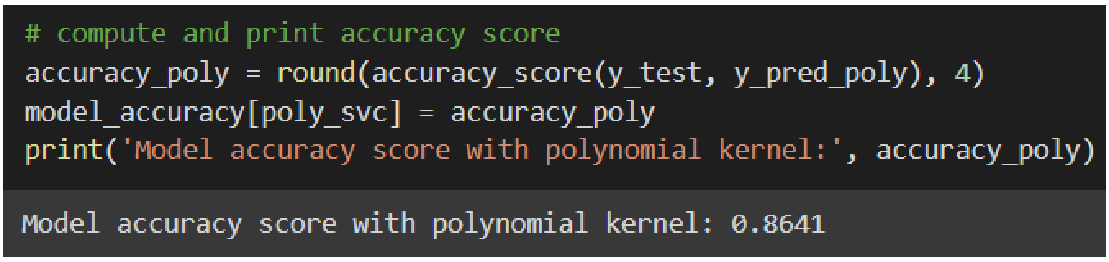
```
# compute and print accuracy score
accuracy_svc = round(accuracy_score(y_test, y_pred_svc), 4)
model_accuracy = {}
model_accuracy[svc] = accuracy_svc
print('Model accuracy score with default hyperparameters (rbf kernel):', accuracy_svc)
```

```
Model accuracy score with default hyperparameters (rbf kernel): 0.8706
```



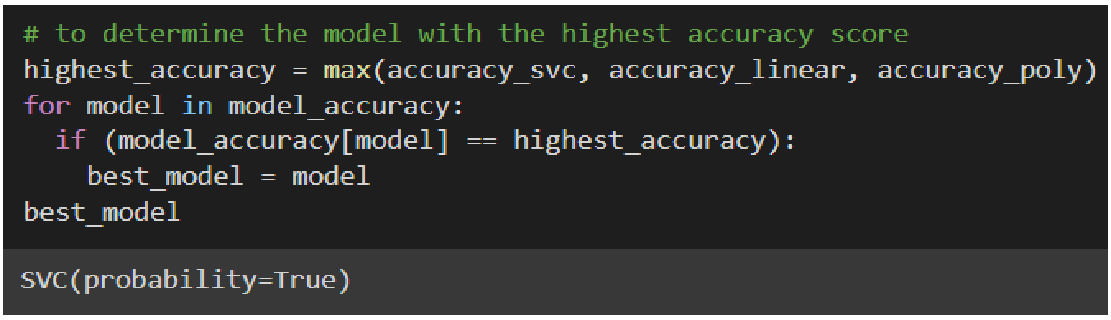
```
# compute and print accuracy score
accuracy_linear = round(accuracy_score(y_test, y_pred_linear), 4)
model_accuracy[linear_svc] = accuracy_linear
print('Model accuracy score with linear kernel:', accuracy_linear)
```

```
Model accuracy score with linear kernel: 0.8681
```



```
# compute and print accuracy score
accuracy_poly = round(accuracy_score(y_test, y_pred_poly), 4)
model_accuracy[poly_svc] = accuracy_poly
print('Model accuracy score with polynomial kernel:', accuracy_poly)
```

```
Model accuracy score with polynomial kernel: 0.8641
```

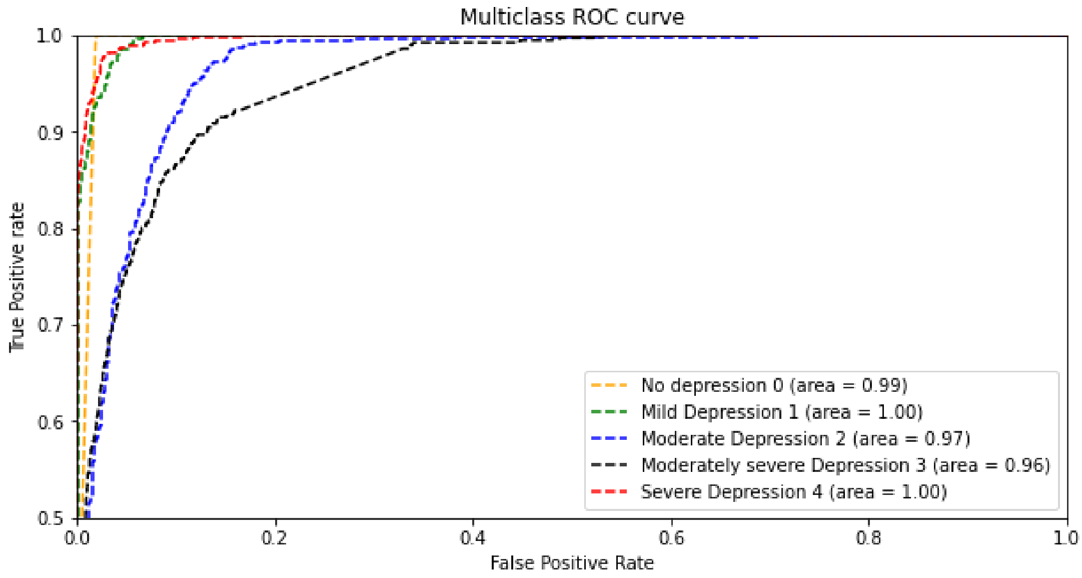


```
# to determine the model with the highest accuracy score
highest_accuracy = max(accuracy_svc, accuracy_linear, accuracy_poly)
for model in model_accuracy:
  if (model_accuracy[model] == highest_accuracy):
    best_model = model
best_model
```

```
SVC(probability=True)
```

Initially, three support-vector classifiers with different kernels were trained, the RBF kernel (first picture), the linear kernel (second picture) as well as the polynomial kernel (third picture). We used the accuracy_score() method in the sklearn library to compute the accuracy score for each model. Then, we compare the accuracy score of each model. The best model is the SVC with RBF kernel since it has the highest accuracy score, 0.8706. Eventually, we save the model and load the model to predict the probability and perform classification based on the users' inputs.

2) Area under ROC Curve

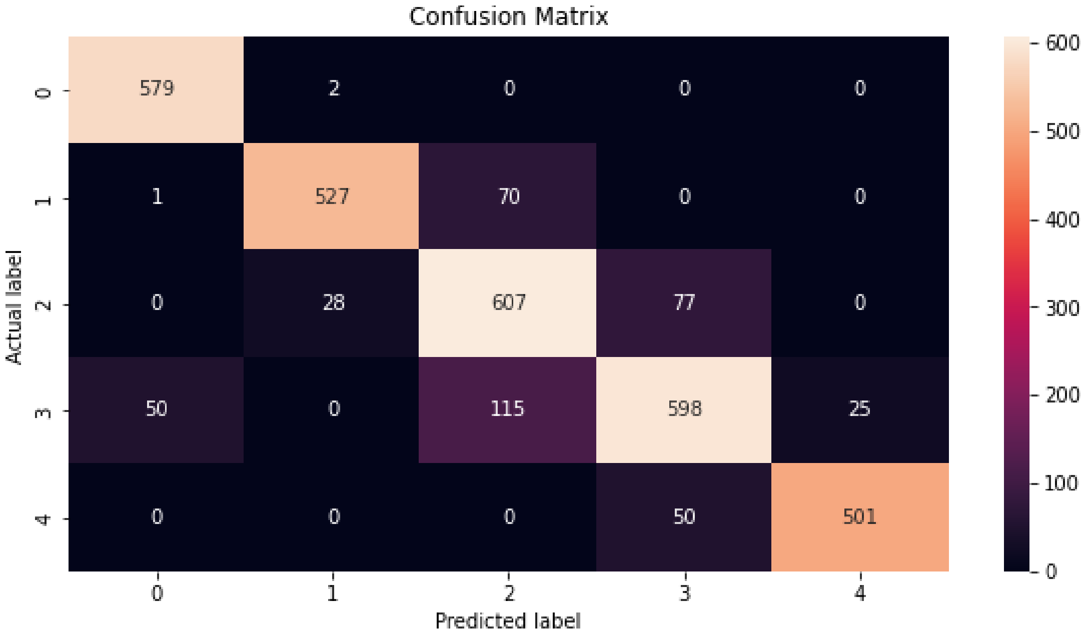


The figure above shows the multiclass ROC curve generated by the SVC RBF kernel model. Normally, the AUC-ROC curve is only used for binary classification problems, but it can also be extended to multiclass classification problems using the One vs All technique. For example, say there are 5 classes (0, 1, 2, 3, 4), the ROC for class 0 will be generated as classifying 0 against the class other than 0 (1, 2, 3, 4). The same thing applies to class 1, class 2, class 3, and class 4.

A ROC curve is constructed by plotting the true positive rate (TPR) against the false positive rate (FPR). TPR is the proportion of actual positives which are correctly predicted, it can be calculated by using the formula: TP/TP+FN. On the other hand, FPR is the ratio between the number of negatives wrongly predicted as positive and the total number of actual negatives, it can be calculated by using the formula: FP/FP+TN.

Based on the ROC curve above, we can see that all of the curves are close to the top-left corner, indicating our model has good performance. The area under curve for the ‘No depression’ class is 0.99, ‘Mild Depression’ class and ‘Severe Depression’ class are having the area under ROC curve of 1, ‘Moderately Depression’ class has the area under ROC curve of 0.97 while the ‘Moderately severe Depression’ has the lowest area under ROC curve among all the classes which is 0.96. Since the areas under ROC curve for all the classes are very close to 1, we can conclude that the SVC classifier that we trained is able to distinguish between all the positive and negative class points correctly. Our model has the capability to predict the probability of each depression severity and classify users into the most possible depression severity with a low error rate.

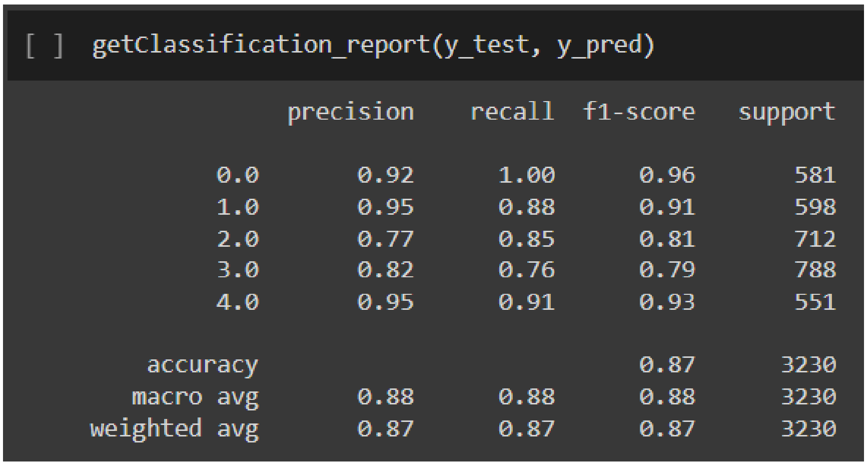
3) Confusion Matrix



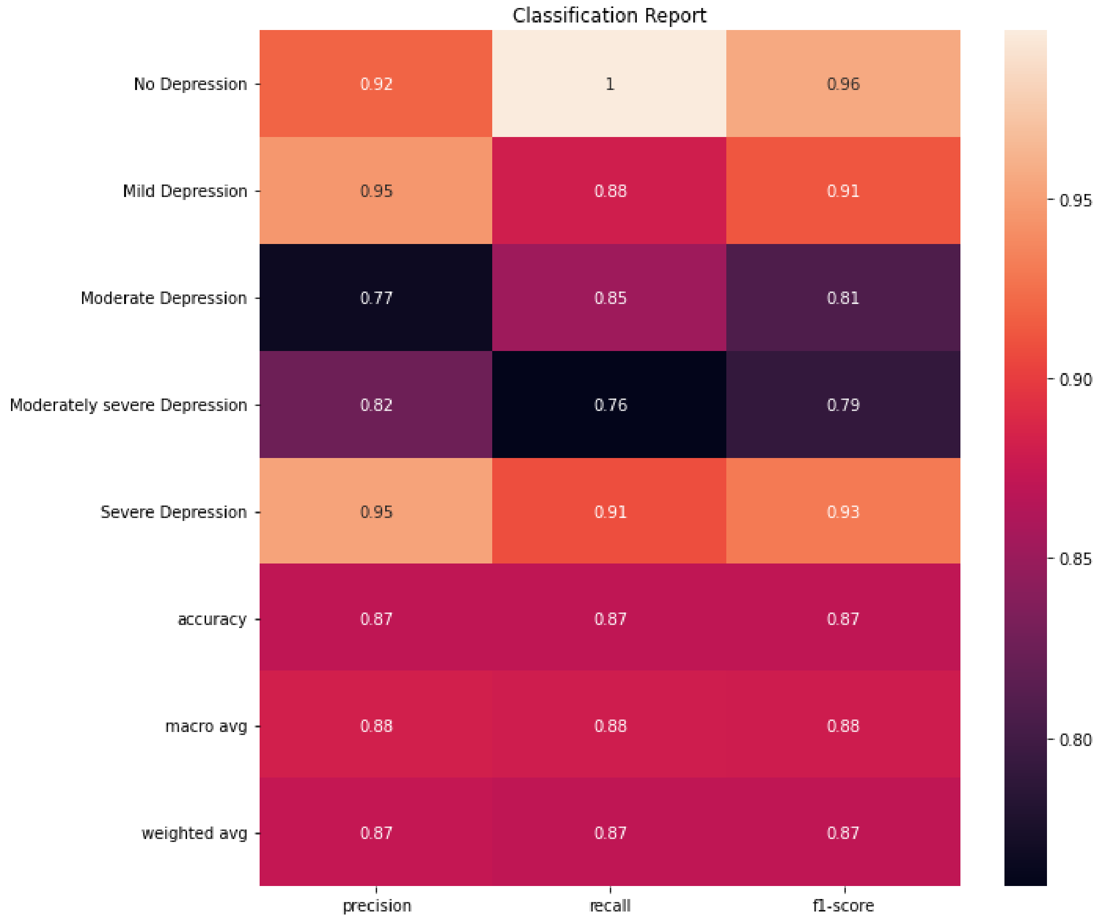
	0	1	2	3	4
0	579	2	0	0	0
1	1	527	70	0	0
2	0	28	607	77	0
3	50	0	115	598	25
4	0	0	0	50	501

The figure above is the confusion matrix for our SVC RBF kernel model. We use a heatmap to visualize the confusion matrix so that it is more comprehensive. The diagonal of the table shows the number of outputs that are predicted correctly which is also known as true positive for the particular row and column. Since we are doing multiclass classification instead of binary classification, we will not get the values of TP, TN, FP and FN directly, we need to calculate them for each class. The x-axis represents the predicted label while the actual label is on the y-axis. In multiclass classification, FN will be the sum of values of corresponding rows except for the TP values, FP for a class will be the sum of values of the corresponding column except for the TP value, TN will be the sum of values of all columns and rows except the values of that class which we are calculating the values for and TP will be the value where actual value and predicted value are the same. Therefore, by having all this information, we can further calculate the precision, recall, accuracy, F1 score and so on to further evaluate the model.

4) Classification Report (Precision, Recall, F1 Score)



```
[ ]  getClassification_report(y_test, y_pred)

                  precision    recall  f1-score   support

             0.0       0.92      1.00      0.96       581
             1.0       0.95      0.88      0.91       598
             2.0       0.77      0.85      0.81       712
             3.0       0.82      0.76      0.79       788
             4.0       0.95      0.91      0.93       551

        accuracy                           0.87      3230
       macro avg       0.88      0.88      0.88      3230
    weighted avg       0.87      0.87      0.87      3230
```



The first picture on top shows the screenshot of the classification report generated by using the sklearn library. It basically includes the precision, recall, F1 score and support for each class, it also computes the accuracy and average value using the precision, recall and F1 score. As resulted in the classification report, our model has recorded high scores in all the aspects, including precision, recall and F1 score. This proves that the overall performance of our model is good, it is consistent and it can make predictions accurately. To ease the performance analysis process, we use a heatmap to visualize the classification report (second picture above).

Suggestions for Future Works

1. **Get more data**

   Adding more data would allow the “data to speak for itself”, instead of relying on assumptions and weak correlations. More data would result in better and more accurate models. To expand the sample size, new data either in the form of new cases or features can be presented. A large variety of data that encompasses a wide range of scenarios can train the model to avoid making biased decisions.

2. **Improve Performance with Ensembles**

   To further increase the accuracy of an individual SVM, we can use an ensemble of SVMs, composed of classifiers that are as accurate and divergent as possible. For example, numerous models created using the same or different techniques are created and the mean or mode of the well-performing model is taken and combined. Alternatively, we can combine data samples by creating numerous subsamples of the training data and train a high-performing algorithm, then combine forecasts.

3. **Regularization**

   Regularization refers to the techniques used to calibrate models in order to minimize the adjusted loss function and prevent overfitting or underfitting. It is intended to reduce a model’s generalization error but not its training error. We can think of regularization as a penalty against complexity. Our goal is to prevent our model from memorizing the training dataset, instead, we want a model that generalizes well to new, unseen data.

Appendix

Link to source code:
https://drive.google.com/drive/folders/182SuymLm5PIOA7HAJZLfKDzyp90pDzGr?usp=sharing

Link to presentation slides:
https://docs.google.com/presentation/d/1mum5JwMtw_H2awDZLIm7z53JimhjTB5-/edit?usp=sharing&ouid=100492958016660987624&rtpof=true&sd=true

Link to presentation video:
https://drive.google.com/file/d/1peb6cr_r0ywqQuAQ5E9zRtXNqTb6Sa2y/view?usp=sharing