1 Understanding the Dataset

1.1 Explore the given Stress-Predict Dataset and understand the problem statement:

The main goal here is to understand the stress-predict Dataset and identify which variables are dependent and which variables are independent. Moreover, we have to use an algorithm in order to predict if the participants are stressed or not. In addition to this also perform necessary pre-processing steps.

1.2 Select an appropriate machine learning category (e.g., classification) and briefly explain why it is suitable for solving this problem:

For the given dataset we can use a category in machine learning called as classification. Classification will be best fit, as after exploring the dataset, one can observe that the target variable is categorical/discrete variable i.e the value of label is either 0 or 1.

2 Data Exploration

2.1 Analyze and report the distribution of data between training and testing sets

In-order to avoid over-fitting, the data should be divided into training and testing set. The optimal ratio to split the data so that we can achieve a good accuracy is 70-80 percent of the data can be used for training and 20-30 percent of the data towards testing. Here, 70 percent of the data is used for training and 30 percent is used for testing.

2.2 Determine whether this dataset can be considered imbalanced. Provide a brief explanation:

The given Dataset can be considered as mildly imbalanced. There are multiple degrees of data imbalance such as mild, moderate and extreme, in the given data set the majority class is having $\sim$ 67.3 percent of the data and the minority class is having $\sim$ 32.7 percent of the data so this can be considered as a mildly imbalanced Dataset. Below is a snippet of a bar plot by which one can determine whether this Dataset is balanced or not.

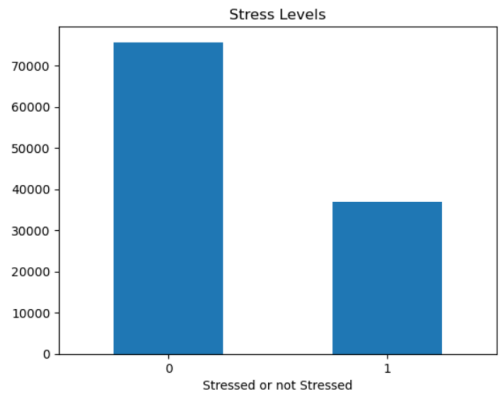

Figure 1: Bar Chart

3 Choosing an ML Package

3.1 Research and list open-source machine learning packages suitable for classification tasks

There are numerous open-source machine learning packages which can be used for classification tasks., ranging from data manipulation to data visualization. Below are some of the packages.

1. **Pandas**
2. **NumPy**
3. **Matplotlib**
4. **ScikitLearn**

3.2 Select one package and provide a short overview of its main features and why you think it's a good choice for this assignment

Scikit-Learn is a good machine learning package for this assignment for number of reasons. Firstly, this library is performance optimized and many classification algorithms such as KNN, Decision Tree etc can be implemented in a highly efficient method. Moreover, it consists of multiple algorithms which can give us the flexibility to choose the algorithm which best fits for us. In addition to this, one of the important feature which this package provides is the splitting of data into training and testing parts.

4 Data Pre-processing

4.1 Prepare the Stress-Predict dataset for input into the chosen ML package:

The following steps were performed in pre-processing or EDA step: -

1. Renaming the columns/attributes.
2. Removing the missing values from the dataset.

Out[29]:

	Participant	Heart_Rate	Respiratory_Rate	Time(sec)	Label
3555	2	NaN	14.231310	1644231138	0
3556	2	NaN	14.188643	1644231139	0
6865	3	NaN	9.682747	1644234689	0
10395	4	NaN	14.054357	1644236927	0
10396	4	NaN	14.022357	1644236928	0

Figure 2: Removing Missing Values

3. Dropping the not required attributes.

Out[4]:

	Heart_Rate	Respiratory_Rate	Label
0	118.00	12.127693	0
1	113.50	12.127693	0
2	93.00	12.127693	0
3	93.25	12.127693	0
4	86.40	12.127693	0

Figure 3: Renaming and Dropping the attributes

4. Splitting the dataset into Training and Testing parts

4.2 Include data pre-processing steps such as handling missing values, feature scaling, and any other necessary transformations, if needed. If not, explain why it is not needed:

Before providing a dataset to any machine learning model the data needs to be cleaned by performing pre-processing steps such as handling missing values, data imbalance checks etc., these steps have been included. These steps are necessary to ensure that the data that is provided to the model is accurate/cleaned so that the model does not learn any wrong information.

5 Algorithm Selection and Application

5.1 Choose two different classification algorithms (e.g., Decision Tree, Random Forest) from the selected ML category:

The two algorithm which have been chosen for this assignment are Decision Tree and Random Forest.

5.2 Apply both algorithms to the pre-processed dataset:

Both the algorithms underwent through necessary preprocessing steps.

5.3 Provide clear descriptions of both algorithms and acknowledge your sources of information

1. **Decision Tree:** It is a non-parametric supervised learning method used for classification and regression tasks [**https://scikit-learn.org/stable/modules/tree.html**]. The decision tree recursively splits the dataset until only leaf nodes are remained i.e., data with only one type of class. DT has two types of nodes; one is the decision node and the other is the leaf node. The decision node contains a condition to split the data and the leaf node decides the class of a new data point. The whole data is first at the root node and then based on the condition the data is again divided into further nodes.
2. **Random Forest:** Random Forest is a kind of ensemble classifier which uses decision tree algorithm in a randomized manner. The first step in random forest is creating a bootstrap data set in a random fashion with the help of sampling. The next step is to select a subset of variables to become the root-node this step is performed at every stage while creating a decision tree (In our case it can either be Heart Rate or Resp. Rate). By following similar steps multiple decision trees are created, after this majority voting takes place in order to predict the target output.

6 Model Evaluation

6.1 Train and test your chosen algorithms using the training set provided (You can divide the data manually or use a built-in function for dividing data in to training and testing set or use the one I provided):

The `train_test_split` method is used which is available in the sckikit learn library to divide the data into training and testing.

6.2 Evaluate the models using appropriate performance metrics (e.g., accuracy, precision, recall, F1-score) on both the training and test sets:

For both the algorithms; accuracy, precision, recall, F1-score and confusion matrix is used to evaluate the performance of the algorithms

6.3 Present your results, including any visualizations or graphics if applicable:

Below is the graphical representation of a decision tree and random forest along with the performance metrices.

1. Decision Tree.

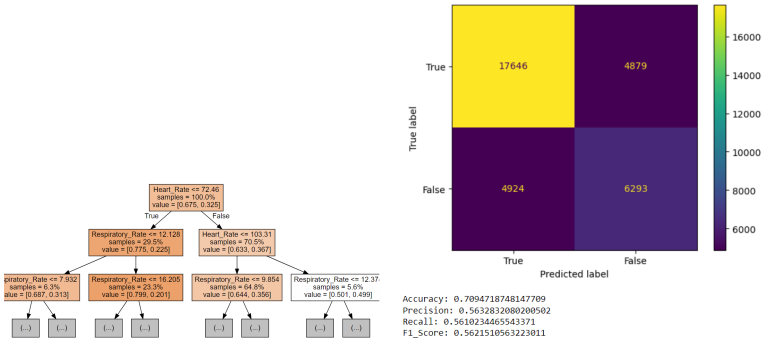

Figure 4: Decision Tree and Confusion Matrix for Decision Tree

2. Random Forest.

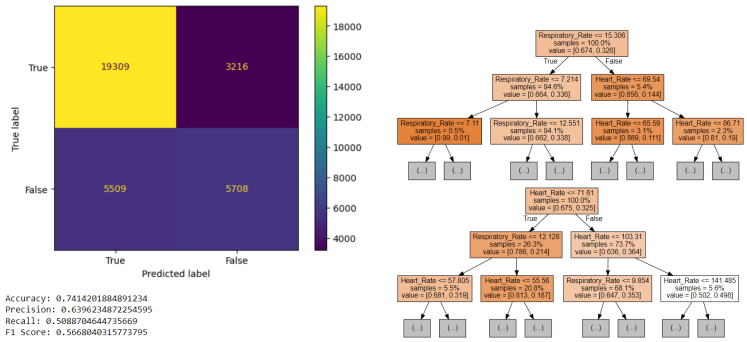

Figure 5: Random Forest and Confusion Matrix for Random Forest

7 Comparative Analysis

7.1 Discuss whether the two models provide similar or significantly different results and explain why:

According to the analysis both the algorithms (Random Forest and Decision Tree) provide almost similar result for the given dataset. But, the precision and accuracy of Random Forest is slightly better than the decision tree this is because random forest selects the average of all the Decision Tree among the generated trees.

7.2 Reflect on the strengths and weaknesses of each algorithm's performance on this dataset:

1. **Strengths of Decision Tree:** 1. The decision tree model is easy to explain. 2. After experimenting on the missing values it was found that the decision tree can handle missing values.
2. **Weakness of decision tree:** It did not returned efficient decision tree for each run.
3. **Strength of Random Forest:** 1. It returned optimal decision tree for each run 2. The accuracy and precision was better as compared decision tree
4. **Weakness of Random Forest:** This algorithm was not able to describe the relationship among the data.