Machine Learning(BITS F464)

Second Semester : 2021-22

Predicting Individuals Mental Health using Machine Learning Methods

By

Group Number : **13**

Vansh Madan - 2018B4A70779H (f20180779@hyderabad.bits-pilani.ac.in)
Shashivardhan Vadyala - 2019A7PS0003H (f20190003@hyderabad.bits-pilani.ac.in)
Dasoju Pranay Kumar - 2019A7PS0006H (f20190006@hyderabad.bits-pilani.ac.in)
Methuku Sheethal Reddy - 2019A7PS0159H (f20190159@hyderabad.bits-pilani.ac.in)
Chakka V Sai Krishna Chaitanya - 2019A7PS017H (f20190171@hyderabad.bits-pilani.ac.in)
Sai Venkata Laxmi Druthi Kommineni - 2019A7PS0023H (f20190023@hyderabad.bits-pilani.ac.in)

Under the supervision of
Prof. Paresh Saxena

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE PILANI

HYDERABAD CAMPUS

April 21, 2022

Acknowledgement

We would like to take this opportunity to express our profound gratitude and deep regards to my mentor Prof. Paresh Saxena and the Course TA Ms. Nida Fatima, for their moral guidance, monitoring, and constant encouragement throughout this project. The blessing, help, and advice given by them time and again shall carry me a long way in the future. Their cordial support, valuable knowledge, and guidance have helped me complete this task through various stages. We are grateful for their cooperation during the period of my assignment. The process of the project was in itself an intimidating task, but after constant and sincere efforts, we came out with flying colors. We hope that the knowledge and the results this project provides will be helpful to one and all.

Abstract

This paper describes a model submitted by our team for the course BITS F464 Machine Learning . The project is about "Predicting Individuals Mental Health using Machine Learning Methods". The dataset used in this study is provided by the ZINDI competition https://zindi.africa/competitions/busaramental health-prediction-challenge/data. The data comes from a survey conducted by the Busara Center in western Kenya. Various machine learning classification models are trained using this dataset. Support Vector Machine, Logistic Regression, Random Forest Classifier, Decision Tree, Gradient Boosting, and Vote Ensembling were all used in our investigations.
On the validation dataset, the SVM, Random Forest, Ada Boosting, and Voting-Ensemble models had the best f1- score and accuracy, with 0.78 and 85 percent, respectively.

Keywords: Support Vector Machine (SVM), Logistic Regression (LR), Decision Tree (DT), Random Forest (RF), Ensemble, Gradient Boosting (GB).

Introduction

Mental health diseases include "depression, bipolar disorder, and other psychoses, dementia, and developmental disorders such as autism," according to the World Health Organization (WHO). The main focus of our project was on predicting depression in Kenyans. Using machine learning approaches, this project aims to predict whether or not a person is going to be depressed. It was decided that evaluating individual behavior using machine learning could aid in better diagnosis and therapy. It helps for having an early interference, which hopefully can result in a fast recovery

The goal of this project is to find a robust, dependable supervised Machine Learning classifier that provides the greatest performance evaluation for predicting whether or not a person is likely to be depressed. The project is based on data collected by the Busara Center in Kenya. SVM, Random Forest, Logistic Regression, and Voting-Ensemble models are among the machine learning approaches evaluated.

Related Work

The problem of predicting mental health has sparked the interest of a huge number of industry and academic researchers recently.

1. **Mental Anxiety and Depression Detection during Pandemic using Machine Learning.**

The four-phase approach suggested in this paper, which includes cluster analysis, NLP, and two psychological screeners, as well as the relationships between the parameters, are both relevant and attainable. The two specific patterns-MU and AU-with greater depressive and anxiety symptom levels are unquestionably at higher risk among the four unique clusters. For those who are most at risk, a realistic diagnosis of mental illness should be made as soon as possible.

2. **Predicting Mental health disorders using Machine Learning for employees in technical and non-technical companies**

The dataset used has over 70 attributes that include both personal and professional information about the employees. In the feature selection stage of data processing, seven out of 70 attributes are chosen as the ones that have the greatest impact on mental illness. Following our research, we discovered that the decision tree classifier had the best performance. With an accuracy of 84 percent and precision of 83 percent, it offers the best accuracy and precision. A history of mental health disorder contributes the most during disorder prediction, followed by family history, according to the feature importance of the selected features.

3. **Predicting Individuals Mental Health Status in Kenya using Machine Learning Methods**

This paper is about predicting individual depression in Kenya, using machine learning methods based on survey data that may help improve diagnostics and have an effective treatment.Random Forest Classifier was used to select the most important features. Results show that SVM, Random Forest, Ada Boosting, and Voting-Ensemble models scored the highest f1- score and accuracy with 0.78 and 85%, respectively, on the validation dataset.

Dataset:

The data includes information about the participants' health, economic activity, financial flows, and household makeup, among other things. The training set has 1143 instances, with 800 cases for the training set and 343 for the validation set after splitting, and 286 cases for the testing set. The dataset is based on a survey conducted by Busara Center in western Kenya.

Data Preprocessing

The data has 75 features including information about economic status, financial flow, etc. To prevent problems arising due to the curse of dimensionality, we conducted feature selection on the data. For this we used the well known embedded technique for feature selection - Random forest importance. Before this, we had to fill missing cells in the data. Since the paper we took for reference didn't mention any specific method filling data, we decided to fill the values with mode of the value.

After this, we ran the random forest importance and reduced the feature space from 75 to 51 with a threshold value of 0.06.

Methodology

We used various classifiers like : Support Vector Machine (SVM) ,Logistic Regression (LR) , Decision Tree, Random Forest Classifier (RF) , Gradient Boosting (GB), Vote Ensembling .

1. SVM

The Support Vector Machine, or SVM, is a popular Supervised Learning technique that may be used to solve both classification and regression problems. The SVM algorithm's purpose is to find the optimum line or decision boundary for categorizing n-dimensional space into classes so that additional data points can be readily placed in the correct category in the future.

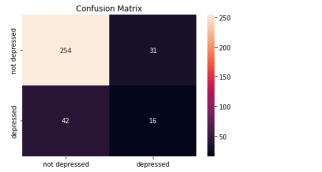
We ran three experiments for SVM with different hyperparameters, kernel, C-parameter, and degree.

SVM_1	Kernel C	Liner 5
SVM_2	Kernel C	RDF 5
SVM_3	Kernel Degree	Poly 2

Result:

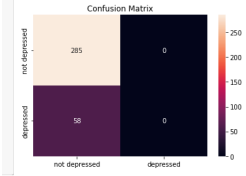
Classification Report for SVM linear model

	precision	recall	f1-score	support
0	0.86	0.89	0.87	285
1	0.34	0.28	0.30	58
accuracy			0.79	343
macro avg	0.60	0.58	0.59	343
weighted avg	0.77	0.79	0.78	343



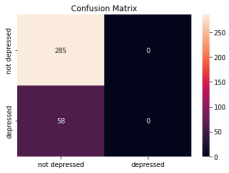
Classification Report for SVM RBF model

	precision	recall	f1-score	support
0	0.83	1.00	0.91	285
1	0.00	0.00	0.00	58
accuracy			0.83	343
macro avg	0.42	0.50	0.45	343
weighted avg	0.69	0.83	0.75	343



Classification Report for SVM Polynomial model

	precision	recall	f1-score	support
0	0.83	1.00	0.91	285
1	0.00	0.00	0.00	58
accuracy			0.83	343
macro avg	0.42	0.50	0.45	343
weighted avg	0.69	0.83	0.75	343



2. Logistic Regression

The Supervised Learning technique includes logistic regression, which is used to predict a categorical dependent variable using a set of independent factors. As a result, the result must be a discrete or categorical value. It can be Yes or No, 0 or 1, true or false, and so on, but instead of giving exact values like 0 and 1, it delivers probabilistic values that are somewhere between 0 and 1. Instead of fitting a regression line as of linear regression, we fit a "S" shaped logistic function in logistic regression, which predicts two maximum values (0 or 1).

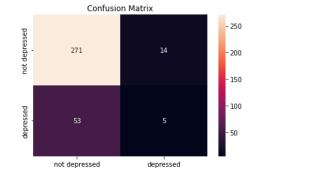
The hyperparameters that were tuned for logistic regression, here, were random state, solver and max_iter.

LG	random_state	0
	solver	liblinear
	max_iter	1000

Result:

Classification Report for Logistic Regression model

	precision	recall	f1-score	support
0	0.84	0.95	0.89	285
1	0.26	0.09	0.13	58
accuracy			0.80	343
macro avg	0.55	0.52	0.51	343
weighted avg	0.74	0.80	0.76	343

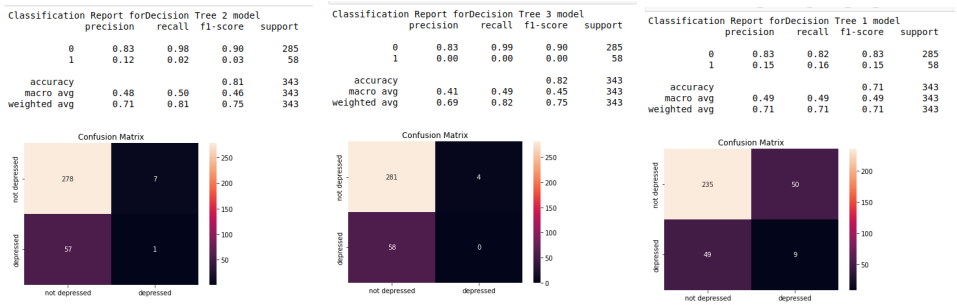


3. Decision Tree

Decision Tree is a supervised learning technique that may be used to solve both classification and regression problems, however it is most commonly employed to solve classification issues. The Decision node and the Leaf node are the two nodes of a tree-structured classifier. Leaf nodes are the output of those decisions and do not contain any more branches, whereas Decision nodes are used to make any decision and have several branches. The decisions or tests are made based on the characteristics of the given dataset. It is a graphical depiction for obtaining all feasible solutions to a problem/decision depending on certain parameters.

First experiment with DT, we only tuned the criterion-hyperparameter and set it to Gini. Second experiment hyperparameters were criterion, max depth, max_leaf_nodes, min_samples_leaf, min_samples_split and splitter. In the third experiment, criterion, max depth, max_leaf_nodes, min_samples_leaf and splitter are the hyperparameters turned.

Result:



4. Random forest

Random Forest is a supervised learning technique that may be used to solve issues in both classification and regression. It is based on ensemble learning, which is a method of integrating several classifiers to solve a complex problem and increase the model's performance. Random Forest is a classifier that combines a number of decision trees on different subsets of a dataset and averages the results to increase the dataset's predicted accuracy. The bigger the number of trees in the forest, the more accurate it is and the problem of overfitting is avoided.

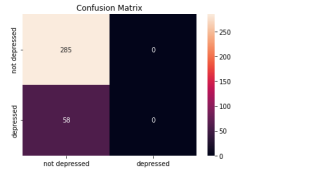
We used the grid search to find the better hyperparameter values. The parameter to be tuned were the criterion, max features, max depth, and n estimators.

RF	criterion	[gini, entropy]
	max_features	[auto, sqrt, log2]
	max_depth	[3,5, 10]
	n_estimators	[100, 120, 150]

Result:

Classification Report for Random forest model

	precision	recall	f1-score	support
0	0.83	1.00	0.91	285
1	0.00	0.00	0.00	58
accuracy			0.83	343
macro avg	0.42	0.50	0.45	343
weighted avg	0.69	0.83	0.75	343



5. Gradient boosting

Gradient boosting is a machine learning technique that can be used for a variety of applications, including regression and classification. It returns a prediction model in the form of an ensemble of weak prediction models, most commonly decision trees. The resulting approach is called gradient-boosted trees when a decision tree is the weak learner; it usually outperforms random forest. A gradient-boosted trees model is constructed in the same stage-wise manner as other boosting approaches, but it differs in that it allows optimization of any *differentiable loss function*.

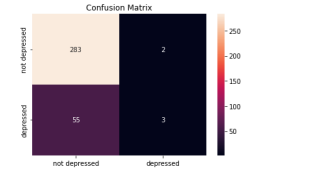
We used grid-search with Gradient boosting to find better hyper-parameter values and the parameters tuned were n_estimators, max_depth, learning_rate, max_features, random_state and criterion.

GB	n_estimators	[10, 20, 50]
	max_depth	[3, 5, 10]
	max_features	[auto, sqrt, log2]
	random_state	[10, 20]
	ciretion	[friedman_mse, mse, mae]

Result:

Classification Report for Gradient Boosting

	precision	recall	f1-score	support
0	0.84	0.99	0.91	285
1	0.60	0.05	0.10	58
accuracy			0.83	343
macro avg	0.72	0.52	0.50	343
weighted avg	0.80	0.83	0.77	343



6. Vote ensembling

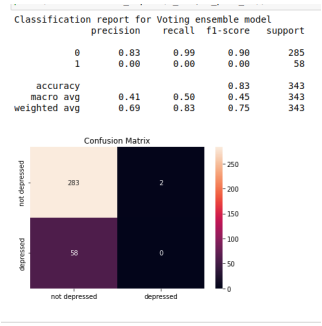
The Voting Classifier is a form of Ensemble Learning that can be homogeneous or heterogeneous, meaning that the basic classifiers can be of the same or different types. As previously stated, this form of ensemble can also be used as a bagging extension (e.g. Random Forest). A Voting Classifier's architecture is made up of a number "n" of machine learning

models, each of which has two types of predictions: hard and soft. The winning prediction in hard mode is the one with "the most votes."

We used five different models: Support Vector Machine, Logistic Regression, Decision Tree, Random Forest, and Gradient Boosting.

Voting: SVM	Kernel C-parapmeter	Rdf 3
Voting: LG	random_state solver max_iter	0 liblinear 1000
Voting: DT	-	default
Voting: RF	-	default
Voting: Ada	-	default

Result:



Proposed Change and after-results:

In data pre-processing, for the data values that were missing, we compiled mode of the available data(for each attribute) and replaced the missing values with calculated mode. But we did not achieve the f1-scores as per the report, so instead we calculated the average of the available data and replaced the missing values with the calculated average. With this, we achieved f1-scores perfectly.

Implementation of Tasks:

Data pre-processing using Random Forest and SVM - Chakka V Sai Krishna Chaitanya

Logistic Regression - Sai Venkata Laxmi Druthi Kommineni

Decision Tree - Dasoju Pranay Kumar

Random Forest - Shashivardhan Vadyala

Gradient Boosting - Methuku Sheethal Reddy

Vote Ensembling - Vansh Madan

Conclusion :

From the above experiments and results, we can see that the Vote - ensembling, Random Forest, and SVM with radial basis kernel yield the best F1 score and accuracy.