Predicting a Person With a Depressive Disorder

Shunya Sunami

1 Introduction

This report outlines the steps we have taken on the CS699 project on predicting a person with a depressive order. The Primary goal of this is to find the best model that can recognize the person with the depressive order based on several factors.

2 Data

The dataset we are using is sourced from a 2020 Behavioral Risk Factor Surveillance System (BRFSS) Survey Data, and modified before being provided. The dataset initially had 5000 tuples with 276 attributes in total.

2.1 Pre-Processing

2.1.1 Removing columns with more than 30 percent of rows being empty

First of all, we removed columns with more than 30 percent of rows being empty,as the columns with the majority of rows being empty does not play a significant role in prediction.

2.1.2 Classification of Remaining Columns

Secondly, we classified the columns to categorical, continuous, and unnecessary. Continuous variables are variables that represent continuous numerical data, such as height, weight, and BMI. Categorical variables, on the other hand, are variables that represent categories or discrete values, such as health status, sleep time, and demographic information. Lastly, unnecessary variables are variables that are deemed unnecessary for the current analysis or are not relevant to the task at hand.

2.1.3 Deletion of Unnecessary Columns

As a next step, we deleted variables that we classified as "unnecessarily". We manually went through the codebook and judged if each variable was useful. One of the removed variables, for example, WEIGHT2 is the variable representing the weight, but there is another attribute for the weight, WTKG3, in the dataset. Similarly, HEIGHT3 and HTIN4 are the attributes for height, but there is another variable for height, HTM4, so those variables are deleted.

2.1.4 Handling Categorical Variables: Filling NAs

After the deletion of the unnecessary columns, we first dealt with the categorical variables. For those attributes with combination number 1 to 9, with 7 and 9 refers to "don't know" and "refused", we found out that majority of those are categorical variables, with some combination of numbers from 1 to 9. We decided to treat NA, 7 and 9 as a same group as 9, since they can be categorized the same. For those having 77 and 99 as "don't know" and "refused" respectively, we replaced 77 with 99 and filled NA with 99.

2.1.5 Handling Continuous Variables: Outlier Detection and Replacement

In this step a function was made to remove the outliers using Z-Score. The formula used was $(x - \bar{x})/\sigma$ (where $\bar{x}$ is the mean and σ is the standard deviation) and the values that were 3 standard deviations away from the mean or whose Z-Score was greater than 3 were referred as outliers. After identifying the outliers, they were replaced by the mean of the column using the mean function in R Programming Language. At the end, a for loop was used to iterate the whole dataset and replace the outliers with the mean of their respective columns.The resulting dataset was saved for further processing.

2.1.6 Handling Continuous Variables: Filling NAs

We first dealt with the numbers that needed to be replaced with 0. For instance, the attribute CHILDREN, which is the number of children, has 88 as "None". Thus, we had to replace it with 0. Similarly, we replaced 88 in PHYSHLTH and 8 in CPDEMO18 with 0. After that we replaced numbers representing "don't know" and 'refused", such as 77, 99, 7 and 9 with NA, then filled them with mean.

2.1.7 Zero Variance

In this step, we tried to find the attributes that have zero variance. We used nearZeroVar function in R to find those attributes, then removed them.

2.1.8 Co-linearity

Finally, we tested if there is any continous variables that have co-linearity. We chose to remove the attributes that have correlation coefficient with more than 0.7.

2.2 Data Balancing

After we spitted into the data into training and test datasets, we applied two different data balancing method to balance the training data. One is Synthetic Minority Oversampling Technique (SMOTE), which generates the synthetic examples of minority class. We over-sampled the "Y" class using this method. The other technique we used was the combination of random over-sampling and random under-sampling. We randomly under-sampled "N" class and oversampled "Y" class to balance out the dataset, using ROSE function in R.

2.3 Attribute Selection

Three different attribute selection methods were applied to each balanced dataset. The techniques we applied were CFS, information gain, and random forest importance. CFS is the correlation based feature selection methods, and it finds the subset of features that are highly correlated with the targeting variable, but have low correlation with each other. Information gain shows the reduction in entropy for each attribute, and we choose attributes having high info gain. Random forest importance uses random forest to find how good each feature is to classify the targeting variable and gives scores of importance. For information gain and random forest importance, we first set high threshold, which led to select only few attributes. We then gradually lowered the threshold depending on the performance of the models.

2.4 Training Datasets

As we explained in the previous section, we used 2 different data balancing techniques and 3 different attribute selection methods. We created the training sets for each combination of those balancing methods and attribute selection methods, thus there are 6 training sets prepared in total.

The combination of methods are as follows:

Training Set 1: SMOTE and CFS
Training Set 2: SMOTE and info gain
Training Set 3: SMOTE and RF importance
Training Set 4: Random over/under sampling and CFS
Training Set 5: Random over/under sampling and info gain
Training Set 6: Random over/under sampling and RF importance

3 Analysis

We fitted 6 different classification models to classify whether a person has a depressive order, and analyze how each model fits. We fitted training set 1 to 6 for each classification model, and named them model 1, model 2,...model 6 respectively.

3.1 Recursive Partitioning Classification

First of all, recursive partitioning classification is fitted. It creates the decision tree for classification, by recursively splitting the dataset based on the variable that best splits the data and its threshold value. This classification model has a hyper-parameter cp, which controls the growth of the tree. We tried 10 different cp-values and built the model, then selected the best model from those. We obtained the confusion matrix and performance measures of each model as follows:

	True N	True Y
Predicted N	694	129
Predicted Y	115	61

Table 1: Confusion Matrix of Model 1

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.86	0.68	0.84	0.86	0.85	0.63	0.18	0.18
Class Y	0.32	0.14	0.35	0.32	0.33	0.63	0.18	0.18
Wt.Average	0.76	0.58	0.75	0.76	0.75	0.63	0.18	0.18

Table 2: Performance Measures of Model 1

	True N	True Y
Predicted N	640	102
Predicted Y	169	88

Table 3: Confusion Matrix of Model 2

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.79	0.54	0.86	0.79	0.83	0.66	0.23	0.22
Class Y	0.46	0.21	0.34	0.46	0.39	0.66	0.23	0.22
Wt.Average	0.73	0.47	0.76	0.73	0.74	0.66	0.23	0.22

Table 4: Performance Measures of Model 2

	True N	True Y
Predicted N	678	105
Predicted Y	131	85

Table 5: Confusion Matrix of Model 3

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.84	0.55	0.87	0.84	0.85	0.64	0.27	0.27
Class Y	0.45	0.16	0.39	0.45	0.42	0.64	0.27	0.27
Wt.Average	0.76	0.48	0.78	0.76	0.77	0.64	0.27	0.27

Table 6: Performance Measures of Model 3

	True N	True Y
Predicted N	623	99
Predicted Y	186	91

Table 7: Confusion Matrix of Model 4

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.77	0.52	0.86	0.77	0.81	0.66	0.22	0.21
Class Y	0.48	0.23	0.33	0.48	0.39	0.66	0.22	0.21
Wt.Average	0.71	0.47	0.76	0.71	0.73	0.66	0.22	0.21

Table 8: Performance Measures of Model 4

	True N	True Y
Predicted N	628	102
Predicted Y	181	88

Table 9: Confusion Matrix of Model 5

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.78	0.54	0.86	0.78	0.82	0.65	0.21	0.21
Class Y	0.46	0.22	0.33	0.46	0.38	0.65	0.21	0.21
Wt.Average	0.72	0.48	0.76	0.72	0.73	0.65	0.21	0.21

Table 10: Performance Measures of Model 5

	True N	True Y
Predicted N	653	108
Predicted Y	156	82

Table 11: Confusion Matrix of Model 6

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.81	0.57	0.86	0.81	0.83	0.64	0.22	0.22
Class Y	0.43	0.19	0.34	0.43	0.38	0.64	0.22	0.22
Wt.Average	0.74	0.50	0.76	0.74	0.75	0.64	0.22	0.22

Table 12: Performance Measures of Model 6

3.2 Logistic Regression

The second classification method we used is logistic regression. It is a one of the statistical methods that can predict the probability of a binary outcomes from several explanatory variables. We used the threshold value of 0.5 in this testing. We obtained the confusion matrix and performance measures of each model as follows:

	True N	True Y
Predicted N	468	75
Predicted Y	341	115

Table 13: Confusion Matrix of Model 1

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.58	0.39	0.86	0.58	0.69	0.63	0.14	0.12
Class Y	0.61	0.42	0.25	0.61	0.36	0.63	0.14	0.12
Wt.Average	0.58	0.40	0.75	0.58	0.63	0.63	0.14	0.12

Table 14: Performance Measures of Model 1

	True N	True Y
Predicted N	544	77
Predicted Y	265	113

Table 15: Confusion Matrix of Model 2

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.67	0.41	0.88	0.67	0.76	0.71	0.22	0.19
Class Y	0.59	0.33	0.30	0.59	0.40	0.71	0.22	0.19
Wt.Average	0.66	0.39	0.77	0.66	0.69	0.71	0.22	0.19

Table 16: Performance Measures of Model 2

	True N	True Y
Predicted N	551	75
Predicted Y	258	115

Table 17: Confusion Matrix of Model 3

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.68	0.39	0.88	0.68	0.77	0.70	0.23	0.21
Class Y	0.61	0.32	0.31	0.61	0.41	0.70	0.23	0.21
Wt.Average	0.67	0.38	0.77	0.67	0.70	0.70	0.23	0.21

Table 18: Performance Measures of Model 3

	True N	True Y
Predicted N	622	85
Predicted Y	187	105

Table 19: Confusion Matrix of Model 4

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.77	0.45	0.88	0.77	0.82	0.70	0.28	0.27
Class Y	0.55	0.23	0.36	0.55	0.44	0.70	0.28	0.27
Wt.Average	0.73	0.41	0.78	0.73	0.75	0.70	0.28	0.27

Table 20: Performance Measures of Model 4

	True N	True Y
Predicted N	609	72
Predicted Y	200	118

Table 21: Confusion Matrix of Model 5

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.75	0.38	0.89	0.75	0.82	0.73	0.31	0.30
Class Y	0.62	0.25	0.37	0.62	0.46	0.73	0.31	0.30
Wt.Average	0.73	0.35	0.79	0.73	0.75	0.73	0.31	0.30

Table 22: Performance Measures of Model 5

	True N	True Y
Predicted N	605	76
Predicted Y	204	114

Table 23: Confusion Matrix of Model 6

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.75	0.40	0.89	0.75	0.81	0.70	0.29	0.28
Class Y	0.60	0.25	0.36	0.60	0.45	0.70	0.29	0.28
Wt.Average	0.72	0.37	0.79	0.72	0.74	0.70	0.29	0.28

Table 24: Performance Measures of Model 6

3.3 Naive Bayes

Naive Bayes is a classification algorithm that is based on Bayes Theorem. It calculates the probability of all estimations given the data and gives an output with higher probability as the estimated result. We obtained the confusion matrix and performance measures of each model as follows:

	True N	True Y
Predicted N	350	68
Predicted Y	459	122

Table 25: Confusion Matrix of Model 1

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.43	0.36	0.84	0.43	0.57	0.70	0.059	0.042
Class Y	0.64	0.57	0.21	0.64	0.32	0.70	0.059	0.042
Wt.Average	0.47	0.40	0.72	0.47	0.52	0.59	0.059	0.042

Table 26: Performance Measures of Model 1

	True N	True Y
Predicted N	541	90
Predicted Y	268	100

Table 27: Confusion Matrix of Model 2

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.67	0.47	0.86	0.67	0.75	0.64	0.16	0.14
Class Y	0.53	0.33	0.27	0.53	0.36	0.64	0.16	0.14
Wt.Average	0.64	0.45	0.75	0.64	0.68	0.64	0.16	0.14

Table 28: Performance Measures of Model 2

	True N	True Y
Predicted N	566	78
Predicted Y	243	112

Table 29: Confusion Matrix of Model 3

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.70	0.41	0.88	0.70	0.78	0.67	0.24	0.22
Class Y	0.59	0.30	0.32	0.59	0.41	0.67	0.24	0.22
Wt.Average	0.68	0.39	0.77	0.68	0.70	0.67	0.24	0.22

Table 30: Performance Measures of Model 3

	True N	True Y
Predicted N	662	107
Predicted Y	147	83

Table 31: Confusion Matrix of Model 4

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.82	0.56	0.86	0.82	0.84	0.70	0.24	0.24
Class Y	0.44	0.18	0.36	0.44	0.40	0.70	0.24	0.24
Wt.Average	0.75	0.49	0.77	0.75	0.75	0.70	0.24	0.24

Table 32: Performance Measures of Model 4

	True N	True Y
Predicted N	647	97
Predicted Y	160	93

Table 33: Confusion Matrix of Model 5

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.80	0.51	0.87	0.80	0.83	0.72	0.26	0.26
Class Y	0.49	0.20	0.37	0.49	0.42	0.72	0.26	0.26
Wt.Average	0.74	0.45	0.77	0.74	0.76	0.72	0.26	0.26

Table 34: Performance Measures of Model 5

	True N	True Y
Predicted N	636	98
Predicted Y	173	92

Table 35: Confusion Matrix of Model 6

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.79	0.52	0.87	0.79	0.82	0.69	0.24	0.23
Class Y	0.48	0.21	0.35	0.48	0.40	0.69	0.24	0.23
Wt.Average	0.73	0.46	0.77	0.73	0.74	0.69	0.24	0.23

Table 36: Performance Measures of Model 6

3.4 Random Forest

Random Forest is an ensemble learning method used for classification and regression tasks. It operates by constructing a multitude of decision trees during training and outputs the mode (classification) or mean prediction (regression) of the individual trees. In the `run_random_forest` function, the `train` function from the `caret` package was employed to perform parameter tuning for the Random Forest model using cross-validation (`method = "cv"`). A grid of hyper parameters (`tune_grid`) was defined to tune the number of variables randomly sampled as candidates at each split (the `mtry` parameter). We obtained the confusion matrix and performance measures of each model as follows:

	True N	True Y
Predicted N	709	141
Predicted Y	100	49

Table 37: Confusion Matrix of Model 1

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.87	0.74	0.83	0.87	0.85	0.56	0.14	0.14
Class Y	0.25	0.12	0.32	0.32	0.29	0.56	0.14	0.14
Wt.Average	0.75	0.62	0.73	0.73	0.74	0.56	0.14	0.14

Table 38: Performance Measures of Model 1

	True N	True Y
Predicted N	687	117
Predicted Y	122	73

Table 39: Confusion Matrix of Model 2

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.85	0.61	0.85	0.84	0.85	0.61	0.23	0.23
Class Y	0.38	0.15	0.37	0..38	0.37	0.61	0.23	0.23
Wt.Average	0.76	0.52	0.76	0.76	0.76	0.61	0.23	0.23

Table 40: Performance Measures of Model 2

	True N	True Y
Predicted N	612	88
Predicted Y	197	102

Table 41: Confusion Matrix of Model 3

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.75	0.46	0.87	0.75	0.81	0.64	0.25	0.24
Class Y	0.53	0.2	0.34	0.53	0.41	0.64	0.25	0.24
Wt.Average	0.71	0,42	0.77	0.71	0.73	0.64	0.25	0.24

Table 42: Performance Measures of Model 3

	True N	True Y
Predicted N	681	114
Predicted Y	128	76

Table 43: Confusion Matrix of Model 4

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.84	0.60	0.85	0.84	0.84	0.62	0.23	0.23
Class Y	0.40	0.15	0.37	0.40	0.38	0.62	0.23	0.23
Wt.Average	0.75	0.51	0.76	0.75	0.76	0.62	0.23	0.23

Table 44: Performance Measures of Model 4

	True N	True Y
Predicted N	687	113
Predicted Y	122	77

Table 45: Confusion Matrix of Model 5

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.84	0.59	0.85	0.84	0.85	0.62	0.25	0.24
Class Y	0.40	0.15	0.38	0.40	0.39	0.62	0.25	0.24
Wt.Average	0.76	0.51	0.76	0.76	0.76	0.62	0.25	0.24

Table 46: Performance Measures of Model 5

	True N	True Y
Predicted N	708	121
Predicted Y	101	69

Table 47: Confusion Matrix of Model 6

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.87	0.63	0.85	0.87	0.86	0.61	0.24	0.24
Class Y	0.36	0.12	0.40	0.36	0.38	0.61	0.24	0.24
Wt.Average	0.77	0.53	0.76	0.77	0.77	0.61	0.24	0.24

Table 48: Performance Measures of Model 6

3.5 Linear Discriminant Analysis

Linear Discriminant Analysis is a classification technique that finds linear combinations of features that best separate different classes in the dataset. It's based on the concept of modeling the distribution of the predictors separately for each class and then using Bayes' theorem to estimate the probability of a sample belonging to a particular class. We obtained the confusion matrix and performance measures of each model as follows:

	True N	True Y
Predicted N	468	75
Predicted Y	341	115

Table 49: Confusion Matrix of Model 1

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.57	0.39	0.86	0.57	0.69	0.62	0.14	0.12
Class Y	0.60	0.42	0.25	0.60	0.35	0.62	0.14	0.12
Wt.Average	0.58	0.40	0.74	0.58	0.62	0.62	0.14	0.12

Table 50: Performance Measures of Model 1

	True N	True Y
Predicted N	546	75
Predicted Y	263	115

Table 51: Confusion Matrix of Model 2

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.67	0.39	0.88	0.67	0.76	0.70	0.22	0.20
Class Y	0.60	0.32	0.30	0.60	0.40	0.70	0.22	0.20
Wt.Average	0.66	0.38	0.76	0.66	0.69	0.70	0.22	0.20

Table 52: Performance Measures of Model 2

	True N	True Y
Predicted N	553	75
Predicted Y	256	115

Table 53: Confusion Matrix of Model 3

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.68	0.39	0.88	0.68	0.77	0.69	0.23	0.21
Class Y	0.60	0.31	0.31	0.60	0.41	0.69	0.23	0.21
Wt.Average	0.66	0.38	0.77	0.66	0.70	0.69	0.23	0.21

Table 54: Performance Measures of Model 3

	True N	True Y
Predicted N	627	87
Predicted Y	182	103

Table 55: Confusion Matrix of Model 4

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.77	0.45	0.87	0.77	0.82	0.70	0.27	0.26
Class Y	0.54	0.22	0.36	0.54	0.43	0.70	0.27	0.26
Wt.Average	0.73	0.41	0.78	0.73	0.75	0.70	0.27	0.26

Table 56: Performance Measures of Model 4

	True N	True Y
Predicted N	616	76
Predicted Y	193	114

Table 57: Confusion Matrix of Model 5

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.76	0.40	0.89	0.76	0.82	0.73	0.30	0.29
Class Y	0.60	0.23	0.37	0.60	0.45	0.73	0.30	0.29
Wt.Average	0.73	0.36	0.79	0.73	0.75	0.73	0.30	0.29

Table 58: Performance Measures of Model 5

	True N	True Y
Predicted N	610	78
Predicted Y	199	112

Table 59: Confusion Matrix of Model 6

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.75	0.41	0.88	0.75	0.81	0.70	0.29	0.27
Class Y	0.58	0.24	0.36	0.58	0.44	0.70	0.29	0.27
Wt.Average	0.72	0.38	0.78	0.72	0.74	0.70	0.29	0.27

Table 60: Performance Measures of Model 6

3.6 Support Vector Machine

Support Vector Machine is a powerful supervised learning algorithm used for classification and regression tasks. Within the `run_svm` function, an Support Vector Machine model was trained using a radial kernel and scaled numeric variables. While hyper parameters such as the choice of kernel (e.g., radial) and scaling method were used . We obtained the confusion matrix and performance measures of each model as follows:

	True N	True Y
Predicted N	499	93
Predicted Y	310	97

Table 61: Confusion Matrix of Model 1

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.61	0.38	0.84	0.61	0.71	0.56	0.10	0.08
Class Y	0.51	0.48	0.23	0.51	0.32	0.56	0.10	0.08
Wt.Average	0.59	0.40	0.72	0.59	0.63	0.56	0.10	0.08

Table 62: Performance Measures of Model 1

	True N	True Y
Predicted N	549	83
Predicted Y	260	107

Table 63: Confusion Matrix of Model 2

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.67	0.32	0.86	0.67	0.76	0.62	0.19	0.17
Class Y	0.56	0.43	0.29	0.56	0.38	0.62	0.19	0.17
Wt.Average	0.65	0.34	0.75	0.65	0.69	0.62	0.19	0.17

Table 64: Performance Measures of Model 2

	True N	True Y
Predicted N	595	81
Predicted Y	214	109

Table 65: Confusion Matrix of Model 3

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.73	0.26	0.88	0.73	0.80	0.65	0.25	0.24
Class Y	0.57	0.42	0.33	0.57	0.42	0.65	0.25	0.24
Wt.Average	0.70	0.29	0.77	0.70	0.72	0.65	0.25	0.24

Table 66: Performance Measures of Model 3

	True N	True Y
Predicted N	649	100
Predicted Y	160	90

Table 67: Confusion Matrix of Model 4

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.80	0.19	0.86	0.80	0.83	0.63	0.24	0.24
Class Y	0.47	0.52	0.36	0.47	0.40	0.63	0.24	0.24
Wt.Average	0.73	0.26	0.77	0.73	0.75	0.63	0.24	0.24

Table 68: Performance Measures of Model 4

	True N	True Y
Predicted N	665	94
Predicted Y	144	96

Table 69: Confusion Matrix of Model 5

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.82	0.17	0.87	0.82	0.84	0.66	0.30	0.29
Class Y	0.50	0.49	0.40	0.50	0.44	0.66	0.30	0.29
Wt.Average	0.76	0.23	0.78	0.76	0.77	0.66	0.30	0.29

Table 70: Performance Measures of Model 5

	True N	True Y
Predicted N	661	102
Predicted Y	148	88

Table 71: Confusion Matrix of Model 6

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.81	0.18	0.86	0.81	0.84	0.64	0.25	0.25
Class Y	0.46	0.53	0.37	0.46	0.41	0.64	0.25	0.25
Wt.Average	0.74	0.25	0.77	0.74	0.75	0.64	0.25	0.25

Table 72: Performance Measures of Model 6

3.7 Model Summary

Model	Model Name	Class N TPR	Class Y TPR
1	Recursive Partitioning 1	0.86	0.32
2	Recursive Partitioning 2	0.79	0.46
3	Recursive Partitioning 3	0.84	0.45
4	Recursive Partitioning 4	0.77	0.48
5	Recursive Partitioning 5	0.78	0.46
6	Recursive Partitioning 6	0.81	0.43
7	Logistic Regression 1	0.58	0.61
8	Logistic Regression 2	0.67	0.59
9	Logistic Regression 3	0.68	0.61
10	Logistic Regression 4	0.77	0.55
11	Logistic Regression 5	0.75	0.62
12	Logistic Regression 6	0.75	0.60
13	Naive Bayes 1	0.43	0.64
14	Naive Bayes 2	0.67	0.53
15	Naive Bayes 3	0.70	0.59
16	Naive Bayes 4	0.82	0.44
17	Naive Bayes 5	0.80	0.49
18	Naive Bayes 6	0.79	0.48
19	Random Forest 1	0.87	0.25
20	Random Forest 2	0.85	0.38
21	Random Forest 3	0.75	0.53
22	Random Forest 4	0.84	0.40
23	Random Forest 5	0.84	0.40
24	Random Forest 6	0.87	0.36
25	Linear Discriminant 1	0.57	0.60
26	Linear Discriminant 2	0.67	0.60
27	Linear Discriminant 3	0.68	0.60
28	Linear Discriminant 4	0.77	0.54
29	Linear Discriminant 5	0.76	0.60
30	Linear Discriminant 6	0.75	0.58
31	Support Vector 1	0.61	0.51
32	Support Vector 2	0.67	0.56
33	Support Vector 3	0.73	0.57
34	Support Vector 4	0.80	0.47
35	Support Vector 5	0.82	0.50
36	Support Vector 6	0.81	0.46

Table 73: TPR for Both Class Across Models

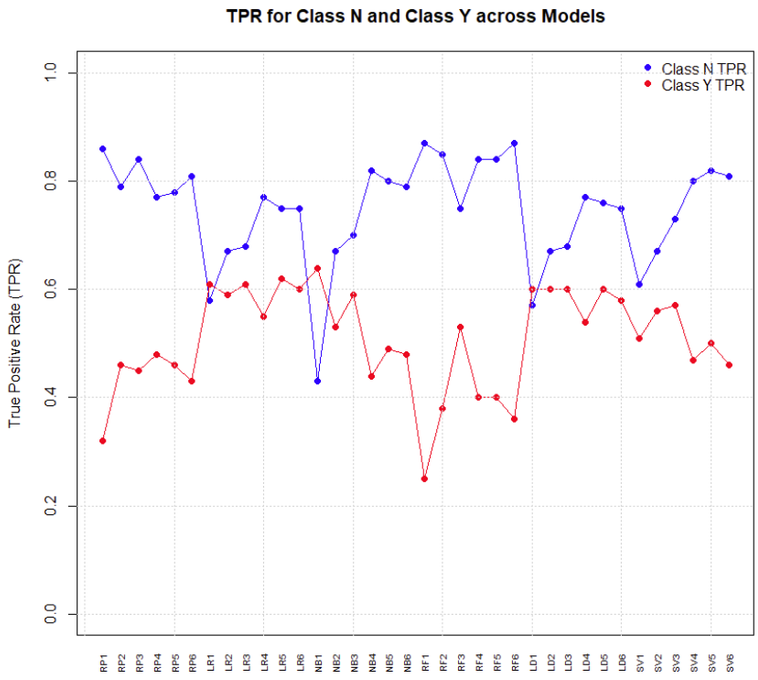

Figure 1: True Positive Rate for Class N and Class Y Across Models

3.8 Choosing Best Model

We have fitted 36 different models in total, and found the performance measures for each model, as shown in the above sections. In choosing the best model, we first shortlisted the models having 0.6 or above class Y TPR and 0.7 or above class N TPR using the Table 73 and Figure 1. Those models are as follows:

- Model 5 of Logistic Regression

	True N	True Y
Predicted N	609	72
Predicted Y	200	118

Table 74: Confusion Matrix of Model 5

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.75	0.38	0.89	0.75	0.82	0.73	0.31	0.30
Class Y	0.62	0.25	0.37	0.62	0.46	0.73	0.31	0.30
Wt.Average	0.73	0.35	0.79	0.73	0.75	0.73	0.31	0.30

Table 75: Performance Measures of Model 5

- Model 6 of Logistic Regression

	True N	True Y
Predicted N	605	76
Predicted Y	204	114

Table 76: Confusion Matrix of Model 6

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.75	0.40	0.89	0.75	0.81	0.70	0.29	0.28
Class Y	0.60	0.25	0.36	0.60	0.45	0.70	0.29	0.28
Wt.Average	0.72	0.37	0.79	0.72	0.74	0.70	0.29	0.28

Table 77: Performance Measures of Model 6

- Model 5 of Linear Discriminant Analysis

	True N	True Y
Predicted N	616	76
Predicted Y	193	114

Table 78: Confusion Matrix of Model 5

	TPR	FPR	Precision	Recall	F-Measure	ROC	MCC	Kappa
Class N	0.76	0.40	0.89	0.76	0.82	0.73	0.30	0.29
Class Y	0.60	0.23	0.37	0.60	0.45	0.73	0.30	0.29
Wt.Average	0.73	0.36	0.79	0.73	0.75	0.73	0.30	0.29

Table 79: Performance Measures of Model 5

Looking at the performance measures, we found that all three models performed similarly well in terms of Precision, Recall, F-Measure, and ROC. However, when we consider Matthews Correlation Coefficient (MCC) and Kappa, Model 5 of Logistic Regression stands out slightly with higher values, suggesting better overall accuracy and agreement.

The parameters of the Model 5 of Logistic Regression are shown in Figure 2, 3, 4, and 5.

4 Discussion

Examining the confusion matrix, particularly for the best-performing model, which is Model 5 of Logistic Regression, provides deeper insights into its classification performance:

	True N	True Y
Predicted N	609	72
Predicted Y	200	118

Table 80: Confusion Matrix of Model 5

```
Coefficients: (13 not defined because of singularities)
             Estimate Std. Error z value Pr(>|z|)
(Intercept)   0.934901   0.652781   1.432 0.152092
X_STATE2     -1.942331   0.730509  -2.659 0.007840 **
X_STATE4     -0.169101   0.332187  -0.509 0.610715
X_STATE5     -0.471082   0.394560  -1.194 0.232500
X_STATE6     -0.166690   0.369767  -0.451 0.652136
X_STATE8     -0.064941   0.307067  -0.211 0.832507
X_STATE9     -0.716100   0.351115  -2.040 0.041400 *
X_STATE10    -1.085971   0.485004  -2.239 0.025149 *
X_STATE11    -0.472873   0.510164  -0.927 0.353976
X_STATE12    -0.670366   0.324194  -2.068 0.038660 *
X_STATE13    -0.658118   0.330637  -1.990 0.046541 *
X_STATE15     0.140667   0.329716   0.427 0.669649
X_STATE16    -0.187641   0.350293  -0.536 0.592187
X_STATE17    -0.945410   0.494245  -1.913 0.055769 .
X_STATE18    -0.190981   0.335382  -0.569 0.569055
X_STATE19    -0.452206   0.354850  -1.274 0.202537
X_STATE20    -1.346456   0.338013  -3.983 6.79e-05 ***
X_STATE21     1.044204   0.379314   2.753 0.005907 **
X_STATE22    -0.238365   0.380299  -0.627 0.530800
X_STATE23     0.101127   0.305932   0.331 0.740982
X_STATE24    -0.422924   0.300716  -1.406 0.159608
X_STATE25    -0.218845   0.335174  -0.653 0.513802
X_STATE26    -0.534866   0.322716  -1.657 0.097441 .
X_STATE27    -0.179624   0.282480  -0.636 0.524851
X_STATE28    -0.674258   0.381668  -1.767 0.077294 .
X_STATE29    -0.422568   0.308538  -1.370 0.170818
X_STATE30     0.120636   0.345154   0.350 0.726703
X_STATE31    -0.021799   0.293534  -0.074 0.940801
X_STATE32    -1.175966   0.530086  -2.218 0.026525 *
X_STATE33    -0.254662   0.398242  -0.639 0.522519
X_STATE34    -0.292626   0.306956  -0.953 0.340429
X_STATE35    -1.178257   0.389071  -3.028 0.002459 **
X_STATE36    -0.606376   0.294409  -2.060 0.039433 *
X_STATE37    -0.164443   0.364900  -0.451 0.652240
X_STATE38     0.518438   0.400946   1.293 0.195998
X_STATE39    -0.142126   0.290538  -0.489 0.624712
X_STATE40    -0.067006   0.347980  -0.193 0.847307
```

Figure 2: Parameters of Best Model - 1

```
X_STATE41     0.474419   0.347298   1.366 0.171930
X_STATE42     0.009766   0.347569   0.028 0.977584
X_STATE44    -0.246437   0.370104  -0.666 0.505501
X_STATE45    -0.545808   0.425754  -1.282 0.199850
X_STATE46    -0.549614   0.364925  -1.506 0.132042
X_STATE47     0.121062   0.389827   0.311 0.756141
X_STATE48    -0.501066   0.308343  -1.625 0.104157
X_STATE49    -0.109769   0.319893  -0.343 0.731491
X_STATE50     0.460219   0.366813   1.255 0.209609
X_STATE51    -0.236240   0.311152  -0.759 0.447707
X_STATE53     0.197673   0.298629   0.662 0.508012
X_STATE54    -0.105537   0.363014  -0.291 0.771262
X_STATE55     0.829875   0.352251   2.356 0.018477 *
X_STATE56    -0.692141   0.428522  -1.615 0.106271
X_STATE66    -0.454133   0.494160  -0.919 0.358095
X_STATE72    -1.360514   0.477902  -2.847 0.004415 **
SEXVAR2       0.702809   0.074403   9.446  < 2e-16 ***
GENHLTH2      0.538173   0.104214   5.164 2.42e-07 ***
GENHLTH3      0.693786   0.108852   6.374 1.85e-10 ***
GENHLTH4      0.931079   0.145789   6.386 1.70e-10 ***
GENHLTH5      1.107755   0.217568   5.092 3.55e-07 ***
GENHLTH9      0.274314   0.543439   0.505 0.613717
ASTHMA32     -0.493266   0.179913  -2.742 0.006112 **
ASTHMA39    -13.893197 882.743563  -0.016 0.987443
HAVARTH42    -0.526388   0.083570  -6.299 3.00e-10 ***
HAVARTH49    -0.829496   0.563789  -1.471 0.141214
MARITAL2      0.327857   0.108056   3.034 0.002412 **
MARITAL3      0.054167   0.132403   0.409 0.682458
MARITAL4      0.600484   0.217629   2.759 0.005794 **
MARITAL5      0.124633   0.105727   1.179 0.238472
MARITAL6      0.365577   0.187858   1.946 0.051651 .
MARITAL9     -0.120550   0.447759  -0.269 0.787752
EMPLOY11     -1.106526   0.458278  -2.415 0.015756 *
EMPLOY12     -1.322464   0.472294  -2.800 0.005109 **
EMPLOY13     -1.013581   0.526519  -1.925 0.054222 .
EMPLOY14     -0.845976   0.483422  -1.750 0.080123 .
EMPLOY15     -0.928037   0.482141  -1.925 0.054251 .
EMPLOY16     -1.427987   0.497935  -2.868 0.004133 **
EMPLOY18     -0.611814   0.473661  -1.292 0.196471
EMPLOY19     -0.446319   0.467620  -0.954 0.339858
```

Figure 3: Parameters of Best Model - 2

```
DECIDE1     -11.952906 431.628894  -0.028 0.977907
DECIDE2     -13.327901 431.628876  -0.031 0.975367
DECIDE9     -11.857823 431.629106  -0.027 0.978083
DIFFWALK1    -1.401658 492.343940  -0.003 0.997728
DIFFWALK2    -1.082466 492.343925  -0.002 0.998246
DIFFWALK9     0.209088 492.345270   0.000 0.999661
DIFFALON1    15.365603 236.852749   0.065 0.948274
DIFFALON2    14.795491 236.852703   0.062 0.950191
DIFFALON9    17.233576 236.853918   0.073 0.941997
HIVTST71      0.476095   0.186965   2.546 0.010883 *
HIVTST72      0.119629   0.184041   0.650 0.515686
HIVTST79      0.522525   0.229207   2.280 0.022625 *
X_RFHLTH2           NA         NA      NA       NA
X_RFHLTH9           NA         NA      NA       NA
X_PHYS14D2    0.502148   0.090938   5.522 3.35e-08 ***
X_PHYS14D3    0.356117   0.197358   1.804 0.071165 .
X_PHYS14D9    0.517735   0.228910   2.262 0.023713 *
X_LTASTH12          NA         NA      NA       NA
X_LTASTH19          NA         NA      NA       NA
X_CASTHM12    0.047361   0.203007   0.233 0.815532
X_CASTHM19    1.453105   0.533091   2.726 0.006414 **
X_ASTHMS12          NA         NA      NA       NA
X_ASTHMS13          NA         NA      NA       NA
X_ASTHMS19          NA         NA      NA       NA
X_DRDXAR21          NA         NA      NA       NA
X_DRDXAR22          NA         NA      NA       NA
X_SEX2              NA         NA      NA       NA
X_AGEG5YR2    -0.689312   0.200794  -3.433 0.000597 ***
X_AGEG5YR3    -0.277641   0.196951  -1.410 0.158630
X_AGEG5YR4    -0.689388   0.205787  -3.350 0.000808 ***
X_AGEG5YR5    -1.051781   0.211189  -4.980 6.35e-07 ***
X_AGEG5YR6    -0.542593   0.203609  -2.665 0.007702 **
X_AGEG5YR7    -0.675980   0.196682  -3.437 0.000588 ***
X_AGEG5YR8    -0.890448   0.201997  -4.408 1.04e-05 ***
X_AGEG5YR9    -0.771762   0.198355  -3.891 9.99e-05 ***
X_AGEG5YR10   -1.599654   0.218073  -7.335 2.21e-13 ***
X_AGEG5YR11   -1.743767   0.228396  -7.635 2.26e-14 ***
X_AGEG5YR12   -2.261117   0.256862  -8.803  < 2e-16 ***
X_AGEG5YR13   -2.429312   0.256842  -9.458  < 2e-16 ***
X_AGEG5YR14   -2.040783   0.404862  -5.041 4.64e-07 ***
```

Figure 4: Parameters of Best Model - 3

```
X_SMOKER32    -0.207924   0.207455  -1.002 0.316218
X_SMOKER33     0.072260   0.129379   0.559 0.576495
X_SMOKER34    -0.289788   0.120867  -2.398 0.016504 *
X_SMOKER39     0.112535   0.328056   0.343 0.731572
X_AIDTST41          NA         NA      NA       NA
X_AIDTST42          NA         NA      NA       NA
X_AIDTST49          NA         NA      NA       NA
PHYSHLTH       0.004215   0.005876   0.717 0.473167
SLEPTIM1      -0.003556   0.023764  -0.150 0.881061
```

Figure 5: Parameters of Best Model - 4

- We correctly classified 609 instances of negatives and 118 instances of positives.
- However, we also misclassified 200 positives as negatives and 72 negatives as positives.

This analysis helps us understand where the model excels and where it struggles. For example, the presence of false positives indicates areas where the model could be improved. This misclassification is the worst error in the model, as it diagnoses patients who actually have depressive order as having no symptoms. In summary, while performance metrics give us a broad overview, the confusion matrix delves into specific misclassifications, offering a clearer understanding of the model's strengths and weaknesses. Combining both perspectives provides a more thorough evaluation of the model's effectiveness.

5 Conclusion

In conclusion, we used six different training datasets and fitted six different models to classify whether the person with depressive order; recursive partitioning, logistic regression, naive bayes, random forest, linear discriminant analysis, and support vector machine. After testing 36 models in total, we found that Model 5 of Logistic Regression is the best model we can find so far. The main limitation of this study is that only 36 models are examined in this study, though there are many other possible models and datasets that can be examined. The other limitation is that the data only contains 5000 tuples, thus it is questionable that if the result can be generalized. We hope that further research on the depressive order could be done to obtain the better model.