Company Bankruptcy Prediction Project

Christos Panourgias

Contents

1 Introduction **3**

2 Methods Applied **3**
- 2.1 Data Separation . . . 3
- 2.2 Normalization . . . 3
- 2.3 Training and Evaluation . . . 3

3 Experimental Results **4**
- 3.1 Training without class balancing . . . 4
  - 3.1.1 Class Percentages . . . 4
  - 3.1.2 Linear Discriminant Analysis . . . 5
  - 3.1.3 Logistic Regression . . . 6
  - 3.1.4 Decision Trees . . . 6
  - 3.1.5 K-Nearest Neighbors . . . 7
  - 3.1.6 Naive Bayes . . . 7
  - 3.1.7 Support Vector Machines . . . 8
  - 3.1.8 Multi-Layer Perceptron . . . 8
  - 3.1.9 Overall Results . . . 8
- 3.2 Training with 3-to-1 Class Balance . . . 9
  - 3.2.1 Class Percentages . . . 9
  - 3.2.2 Linear Discriminant Analysis . . . 10
  - 3.2.3 Logistic Regression . . . 10
  - 3.2.4 Decision Trees . . . 11
  - 3.2.5 K-Nearest Neighbors . . . 11
  - 3.2.6 Naive Bayes . . . 12
  - 3.2.7 Support Vector Machines . . . 12
  - 3.2.8 Multi-Layer Perceptron . . . 13
  - 3.2.9 Overall Results . . . 13
- 3.3 Training with 1-to-1 Class Balance . . . 14
  - 3.3.1 Class Percentages . . . 14
  - 3.3.2 Linear Discriminant Analysis . . . 15
  - 3.3.3 Logistic Regression . . . 15
  - 3.3.4 Decision Trees . . . 16
  - 3.3.5 K-Nearest Neighbors . . . 16
  - 3.3.6 Naive Bayes . . . 17
  - 3.3.7 Support Vector Machines . . . 17
  - 3.3.8 Multi-Layer Perceptron . . . 18
  - 3.3.9 Overall Results . . . 18

4 Conclusions **19**

1 Introduction

Due to the capabilities of today's technology, business managers have the option to leverage the data they collect to optimize their business operations and decisions. A very important factor for the success of a business is its financial health, therefore it is of critical importance that a business manager has high quality information regarding the financial course of their business. The purpose of this work is to analyze financial indicators as well as other information from a series of data of Greek companies, so as to successfully predict the bankruptcy or non-bankruptcy of a company. Therefore, a model will be created which, taking as input the appropriate data, will be able to direct business managers to prevent the bankruptcy of their business.

Data preprocessing will be performed and model architectures such as Logistic Regression, Decision Trees, k-Nearest Neighbors, Support Vector Machines, Naive Bayes, Linear Discriminant Analysis and Multi-Layer Perceptrons will be compared. The results will be evaluated mainly based on the metrics specificity, i.e. how many companies are correctly predicted as bankrupt and recall, i.e. how many companies are correctly predicted as non-bankrupt, other classification evaluation metrics will also be calculated, such as F1-Score, Accuracy and Precision so as to create a comprehensive evaluation for each classifier.

2 Methods Applied

2.1 Data Separation

Initially, the features, which will be the inputs of the model, were separated from the data, as well as the target, which is the output of the model. Then, to implement an objective evaluation of the model, it should be tested on data that is independent from the data used to train it. Therefore, data was split into training data which make up 80% of the original data and test data which make up the remaining 20% of the original data.

2.2 Normalization

Each feature has a different range of values, and this can cause problems in calculations if some values are too large or too small, but also in the time that is spend in operations. Based on this rationale, normalization was implemented using MinMax Scaler so that all features have a value range of $[0, 1]$. It is worth noting that the normalization on the features of the test set was performed based on the maximum and minimum value of the training set so that the test set is not affected by the training data.

2.3 Training and Evaluation

Each model is trained using the .fit() function of the Scikit-Learn library of Python. Once the model has been trained using the training set, the .predict()

function is used to predict the values of the training set, so that the model is evaluated on the predictions of data on which it has been trained, and to predict the values of the test set so that the model is evaluated on the predictions of data that have not been exposed to it.

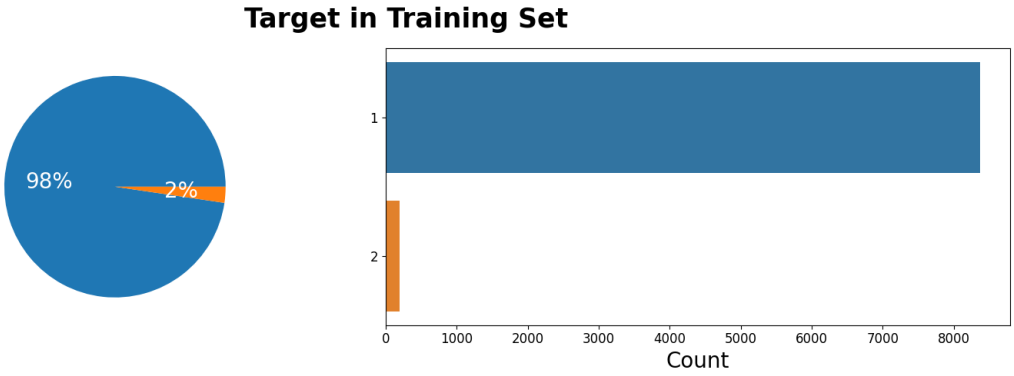
3 Experimental Results

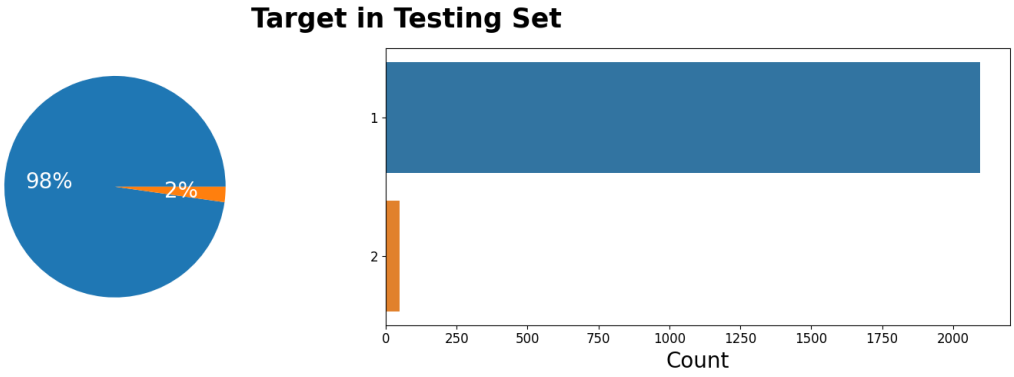
3.1 Training without class balancing

The balance of the classes in the data is an important factor in training the model because it determines the perception the model will have of the importance of each class. Ideally, it is desirable that the model has an equal amount of data from each class, so that it does not make biased predictions about the majority class. The disadvantages of balancing the classes are that it may either be necessary drop data from the majority class, or it may be necessary to create artificial data which may introduce noise into the model.

To understand the need to balance the classes in the training data, the training of the models on the unbalanced training data is presented below.

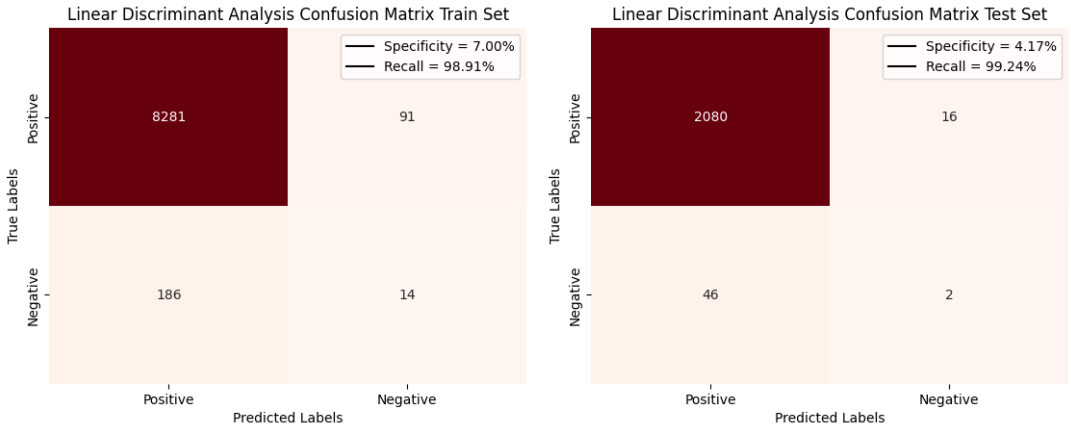
3.1.1 Class Percentages



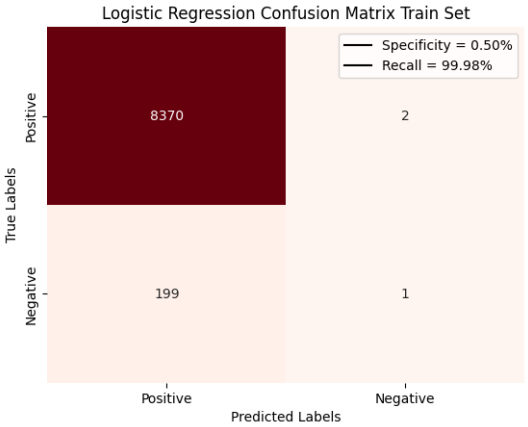


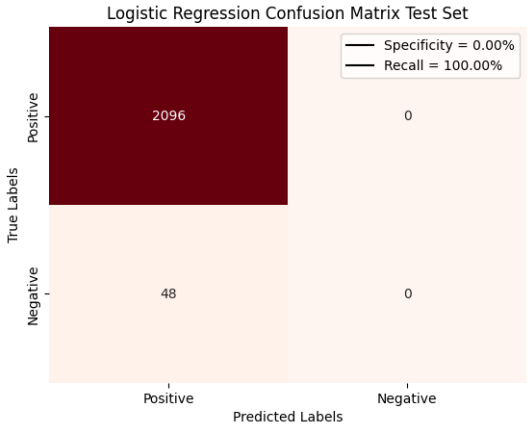
From the graphs above, it is observed that the percentage of the class of bankrupt companies in the data is 2%. If the goal is to predict bankrupt companies then this will make learning difficult for the model. This is also confirmed by the results of the confusion matrices below.

3.1.2 Linear Discriminant Analysis

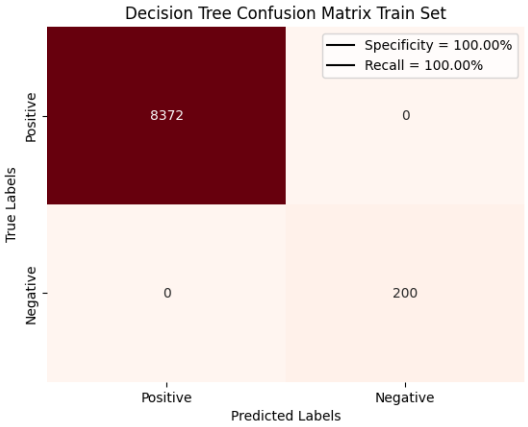


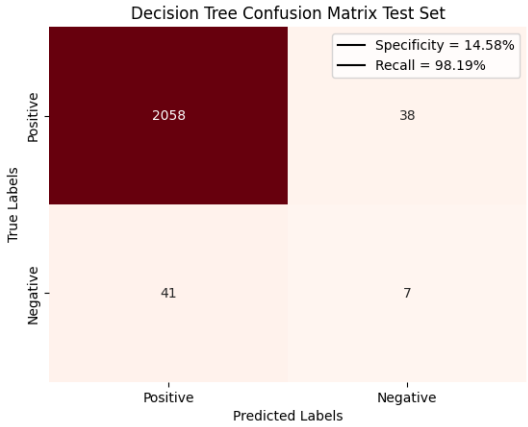
3.1.3 Logistic Regression



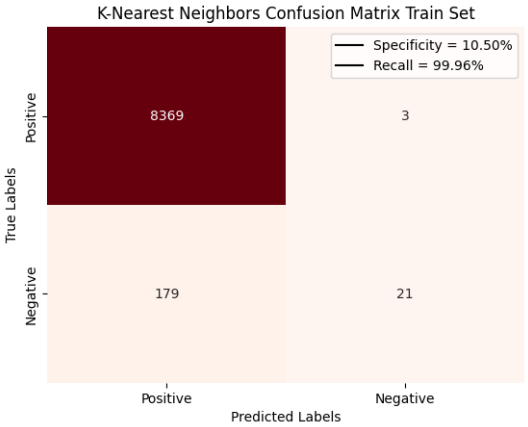


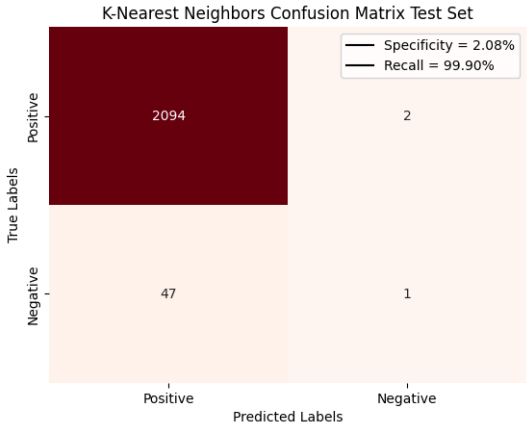
3.1.4 Decision Trees



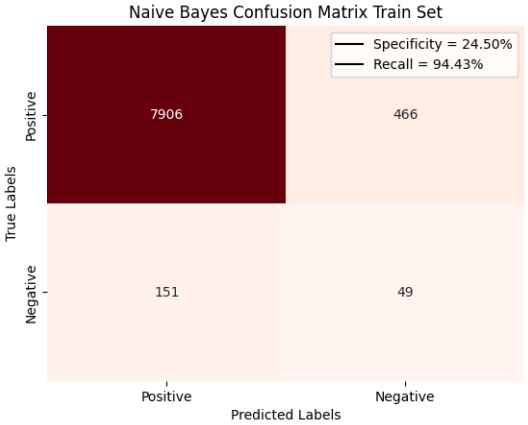


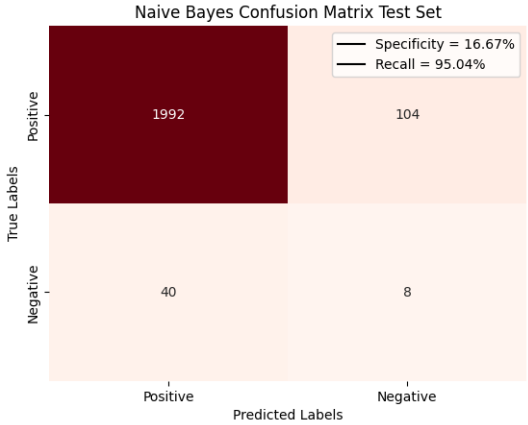
3.1.5 K-Nearest Neighbors



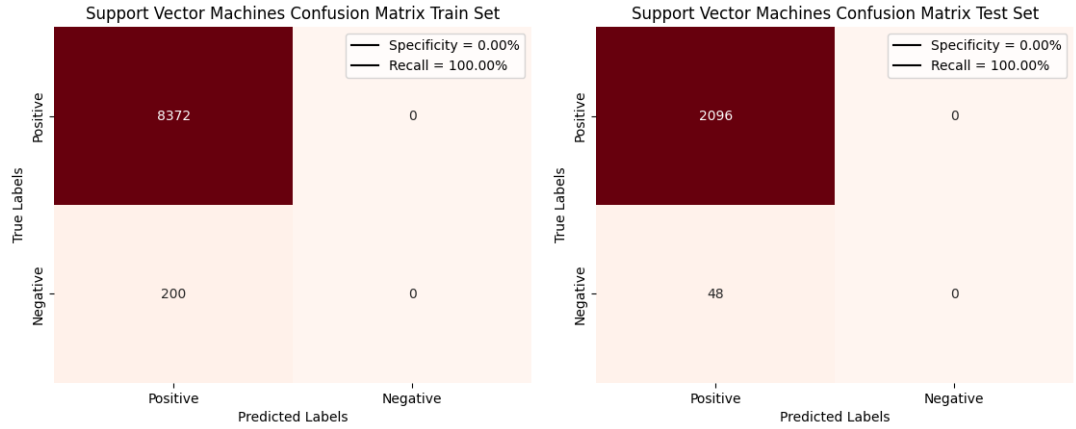


3.1.6 Naive Bayes

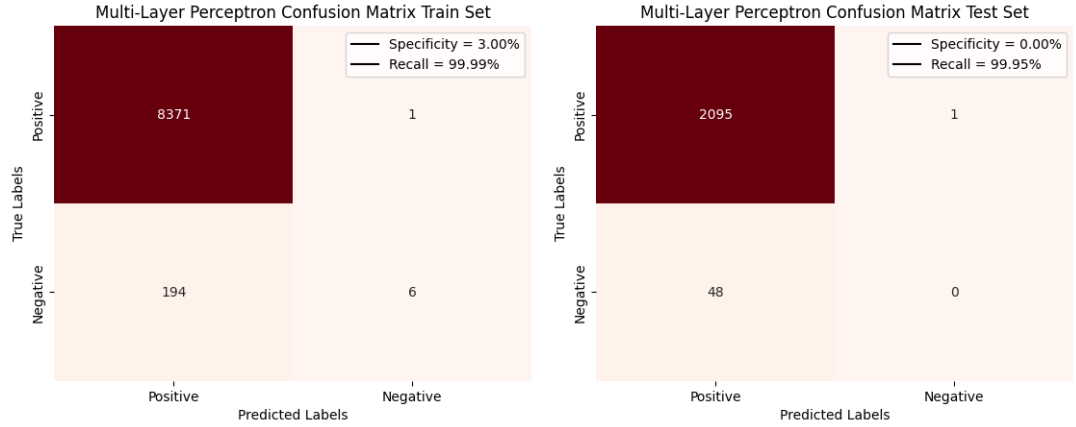




3.1.7 Support Vector Machines



3.1.8 Multi-Layer Perceptron



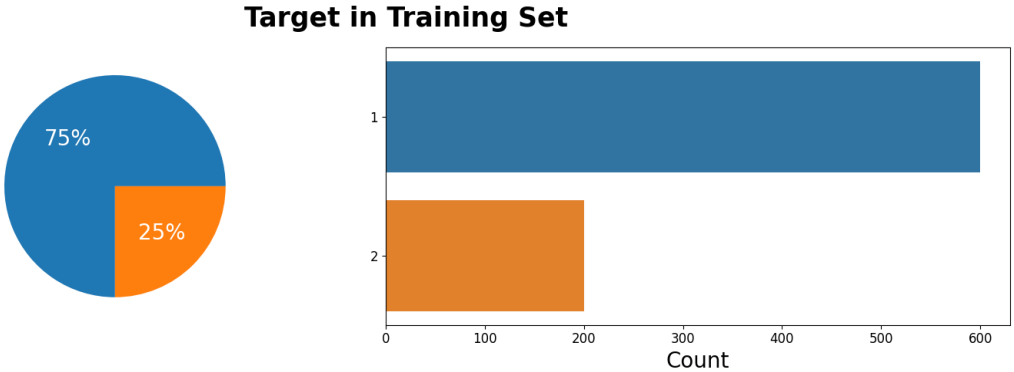
3.1.9 Overall Results

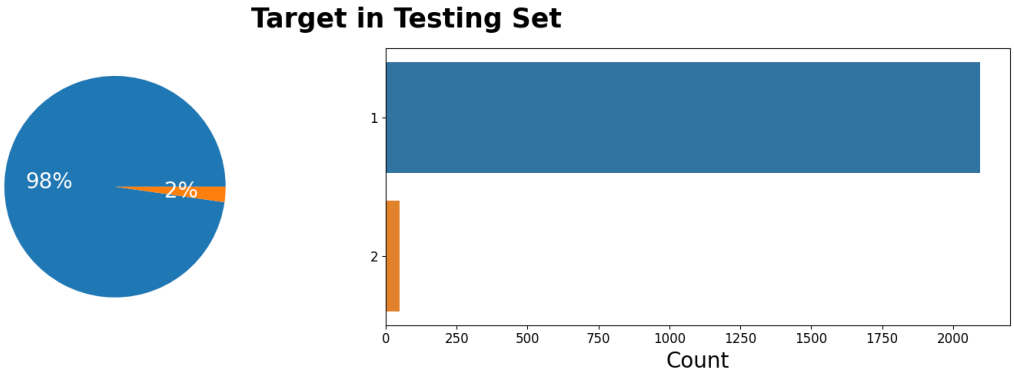
Classifier Name	Set type	Number of training samples	Number of non-healthy companies in training sample	TP	TN	FP	FN	Precision	Recall	Specificity	F1-Score	Accuracy
Linear Discriminant Analysis	Train	8572	200	8281	14	186	91	0.978	0.989	0.07	0.984	0.968
Linear Discriminant Analysis	Test	8572	200	2080	2	46	16	0.978	0.992	0.042	0.985	0.971
Logistic Regression	Train	8572	200	8370	1	199	2	0.977	1	0.005	0.988	0.977
Logistic Regression	Test	8572	200	2096	0	48	0	0.978	1	0	0.989	0.978
Decision Tree	Train	8572	200	8372	200	0	0	1	1	1	1	1
Decision Tree	Test	8572	200	2059	7	41	37	0.98	0.982	0.146	0.981	0.964
K-Nearest Neighbors	Train	8572	200	8369	21	179	3	0.979	1	0.105	0.989	0.979
K-Nearest Neighbors	Test	8572	200	2094	1	47	2	0.978	0.999	0.021	0.988	0.977
Naive Bayes	Train	8572	200	7906	49	151	466	0.981	0.944	0.245	0.962	0.928
Naive Bayes	Test	8572	200	1992	8	40	104	0.98	0.95	0.167	0.965	0.933
Support Vector Machines	Train	8572	200	8372	0	200	0	0.977	1	0	0.988	0.977
Support Vector Machines	Test	8572	200	2096	0	48	0	0.978	1	0	0.989	0.978
Multi-Layer Perceptron	Train	8572	200	8371	3	197	1	0.977	1	0.015	0.988	0.977
Multi-Layer Perceptron	Test	8572	200	2096	0	48	0	0.978	1	0	0.989	0.978

3.2 Training with 3-to-1 Class Balance

From the results of the training on the original data, it seems that no model had $specificity > 62\%$ and $recall > 70\%$ which is what is desired. Below are the results from training the models on balanced data. Specifically, data from the class of companies that belong to the non-bankrupt class were removed so that for every three non-bankrupt companies there is one bankrupt company.

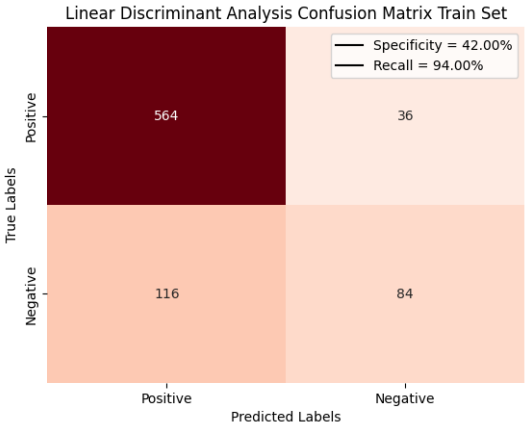
3.2.1 Class Percentages

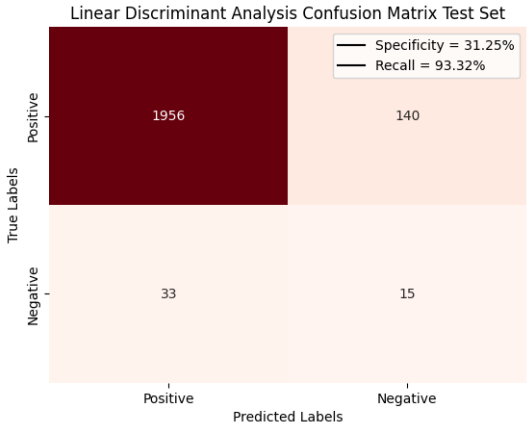




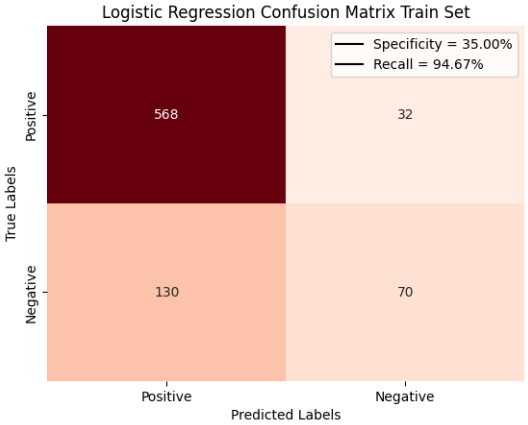
As shown above the training data set is now balanced 3 to 1 and the test set is left as is. Below are the results of training the models on this set.

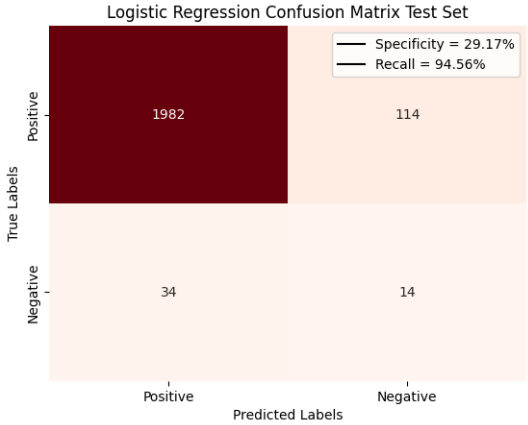
3.2.2 Linear Discriminant Analysis



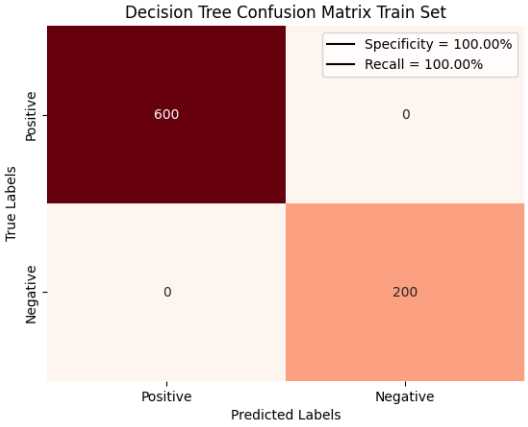


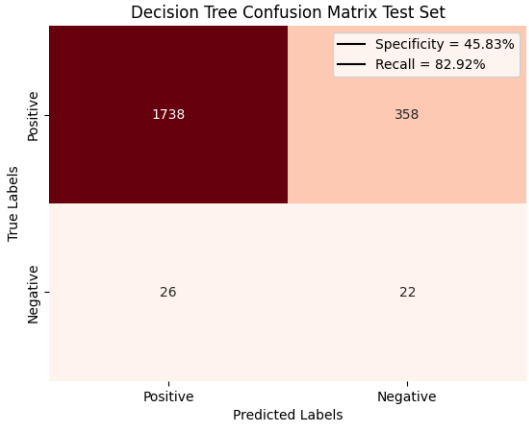
3.2.3 Logistic Regression



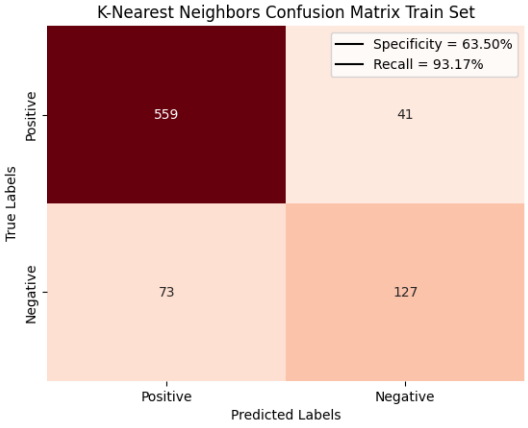


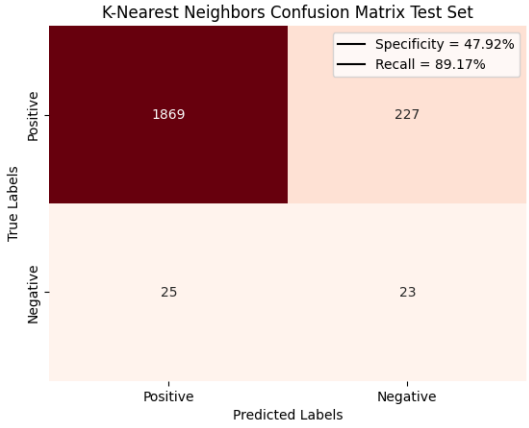
3.2.4 Decision Trees



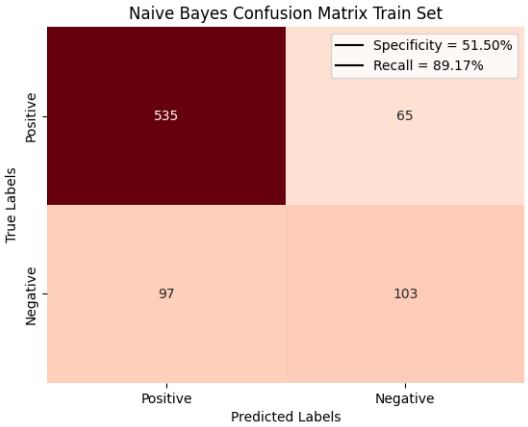


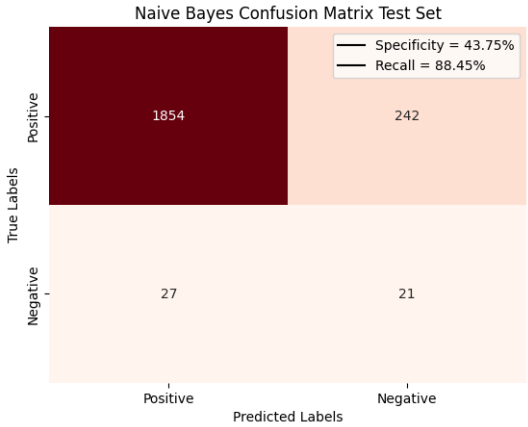
3.2.5 K-Nearest Neighbors



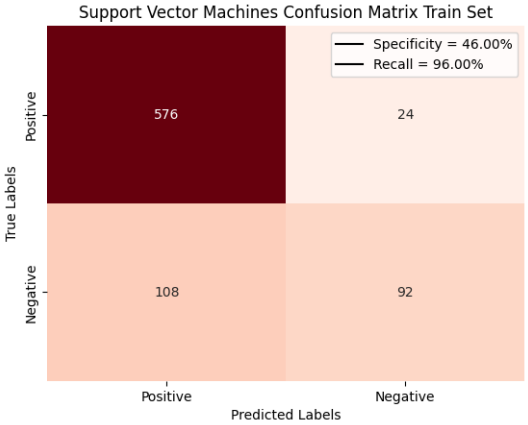


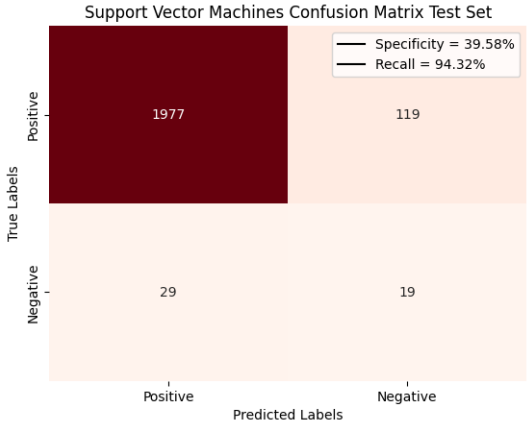
3.2.6 Naive Bayes



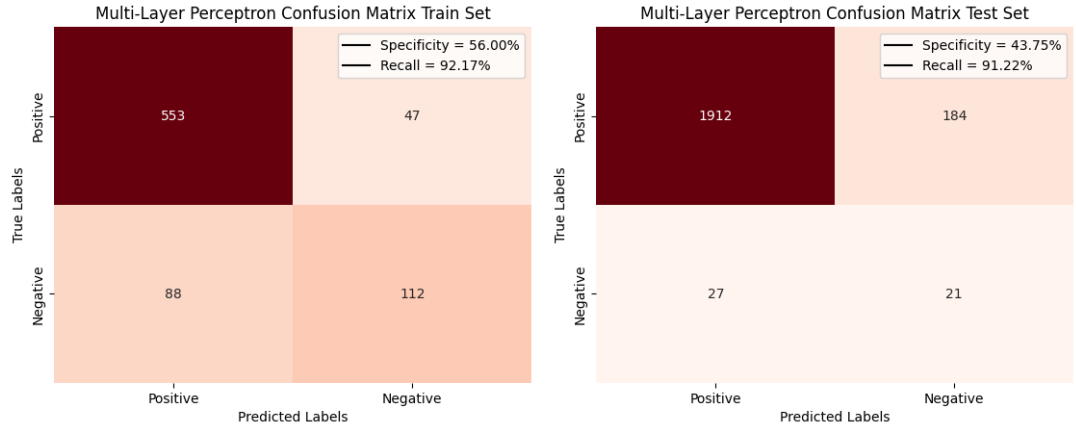


3.2.7 Support Vector Machines





3.2.8 Multi-Layer Perceptron



3.2.9 Overall Results

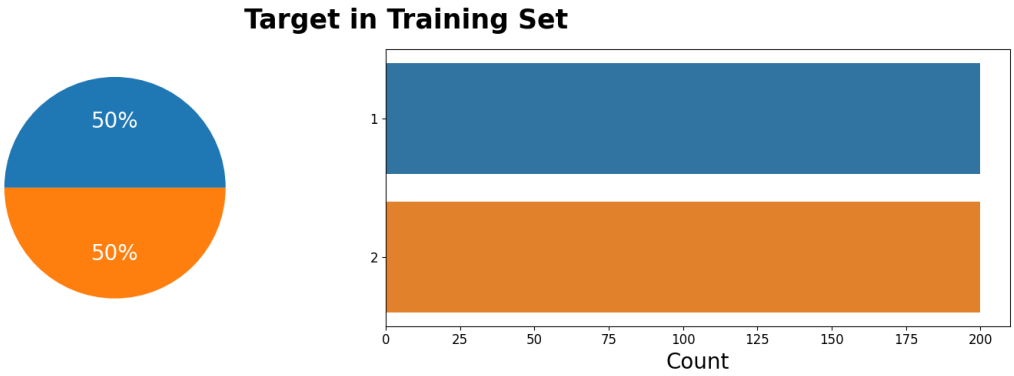
Classifier Name	Set type	Number of training samples	Number of non-healthy companies in training sample	TP	TN	FP	FN	Precision	Recall	Specificity	F1-Score	Accuracy
Linear Discriminant Analysis	Train	800	200	564	84	116	36	0.829	0.94	0.42	0.881	0.81
Linear Discriminant Analysis	Test	800	200	1956	15	33	140	0.983	0.933	0.312	0.958	0.919
Logistic Regression	Train	800	200	568	70	130	32	0.814	0.947	0.35	0.875	0.798
Logistic Regression	Test	800	200	1982	14	34	114	0.983	0.946	0.292	0.964	0.931
Decision Tree	Train	800	200	600	200	0	0	1	1	1	1	1
Decision Tree	Test	800	200	1742	23	25	354	0.986	0.831	0.479	0.902	0.823
K-Nearest Neighbors	Train	800	200	559	127	73	41	0.884	0.932	0.635	0.907	0.858
K-Nearest Neighbors	Test	800	200	1869	23	25	227	0.987	0.892	0.479	0.937	0.882
Naive Bayes	Train	800	200	535	103	97	65	0.847	0.892	0.515	0.869	0.798
Naive Bayes	Test	800	200	1854	21	27	242	0.986	0.885	0.438	0.932	0.875
Support Vector Machines	Train	800	200	576	92	108	24	0.842	0.96	0.46	0.897	0.835
Support Vector Machines	Test	800	200	1977	19	29	119	0.986	0.943	0.396	0.964	0.931
Multi-Layer Perceptron	Train	800	200	557	108	92	43	0.858	0.928	0.54	0.892	0.831
Multi-Layer Perceptron	Test	800	200	1923	21	27	173	0.986	0.917	0.438	0.951	0.907

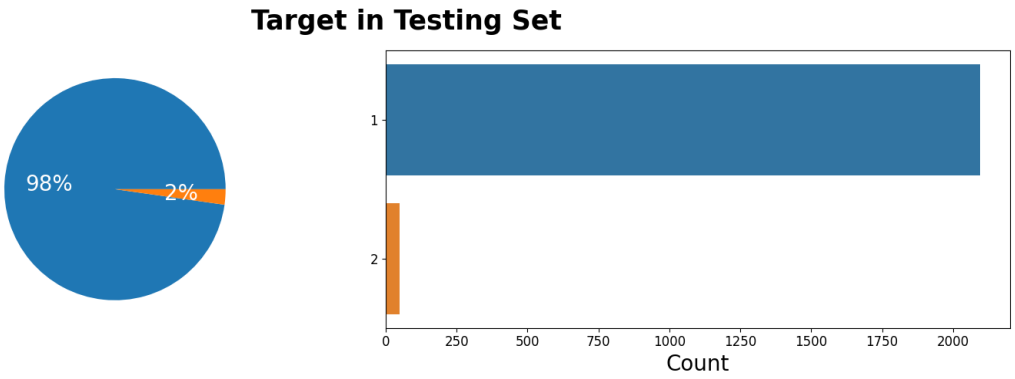
It is observed that none of the models satisfy the success rates required. However, it is worth noting that the percentages of correct prediction as well as the rest of the evaluation metrics increased for each model.

3.3 Training with 1-to-1 Class Balance

Balancing 3-1 improved the results but not enough to keep the model from being biased towards the majority class. Therefore, the classes were balanced 1-1 so that the model considers both classes of equal importance.

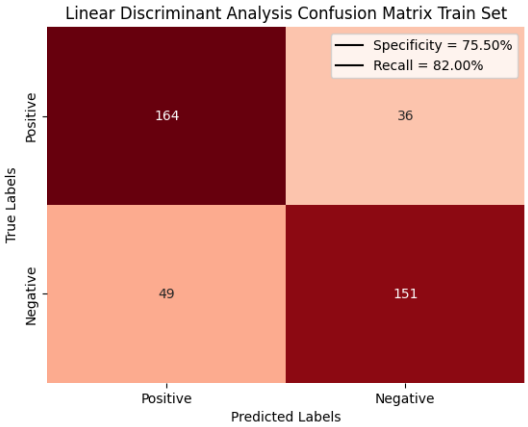
3.3.1 Class Percentages

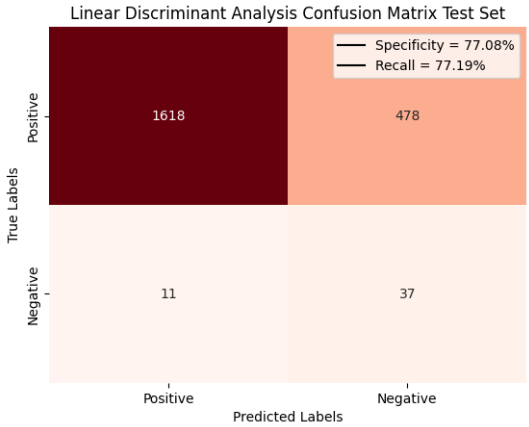




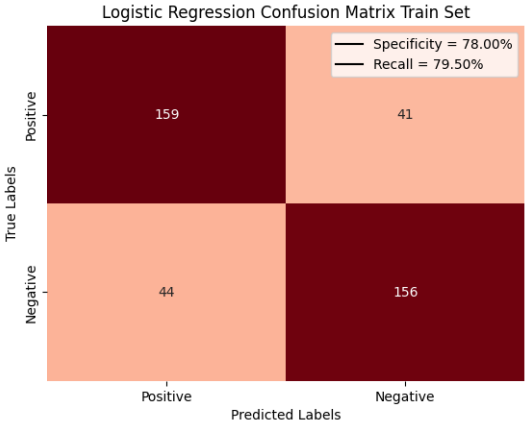
Above is the 1-to-1 balancing of classes in the training set, and below are the confusion matrices and the results of training the models using this set.

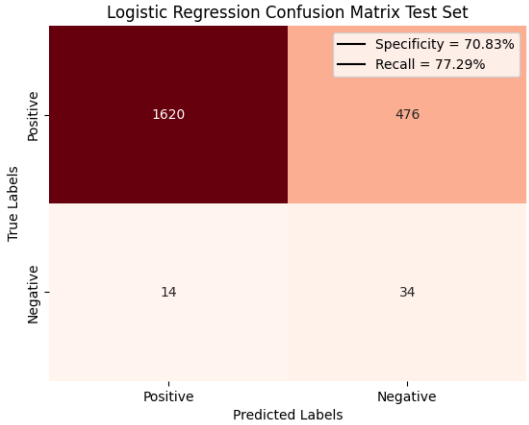
3.3.2 Linear Discriminant Analysis



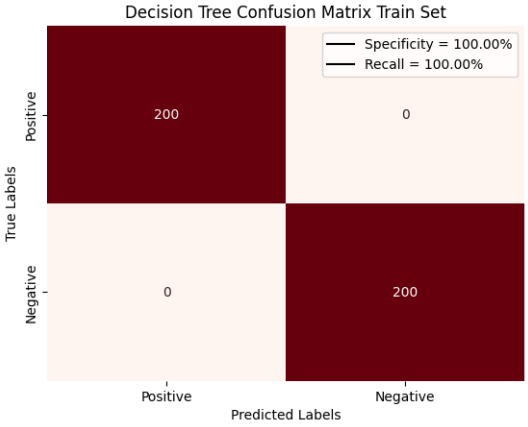


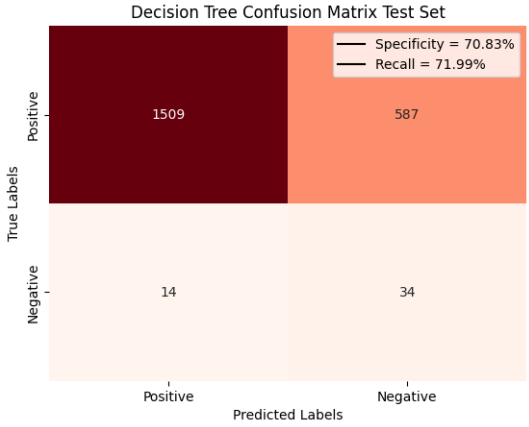
3.3.3 Logistic Regression



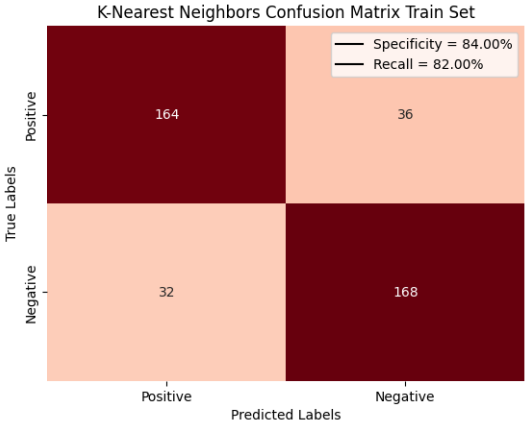


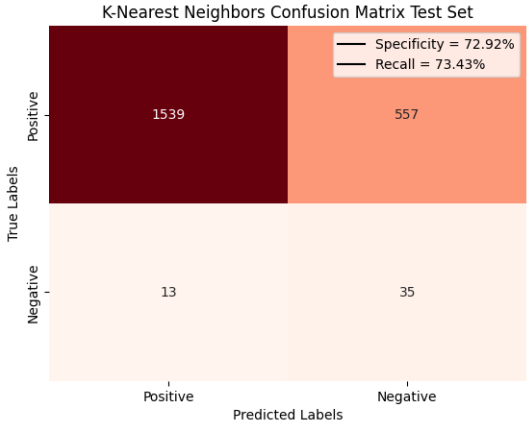
3.3.4 Decision Trees



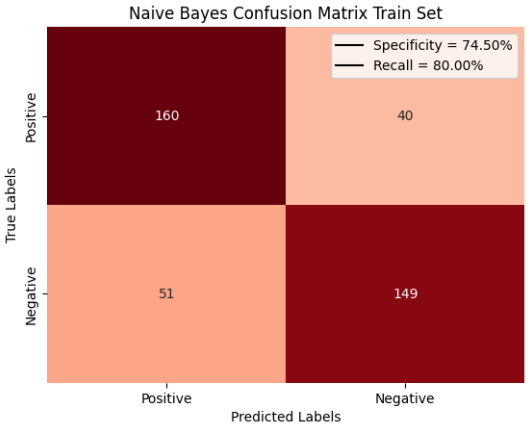


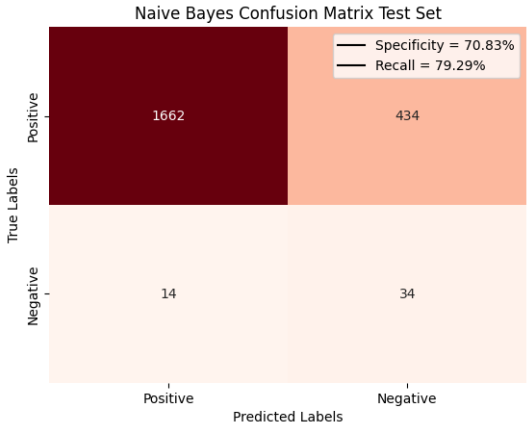
3.3.5 K-Nearest Neighbors



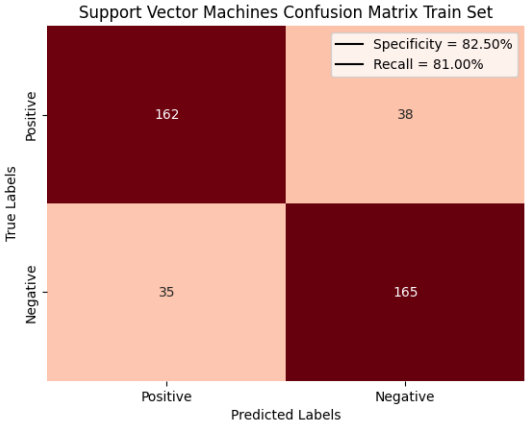


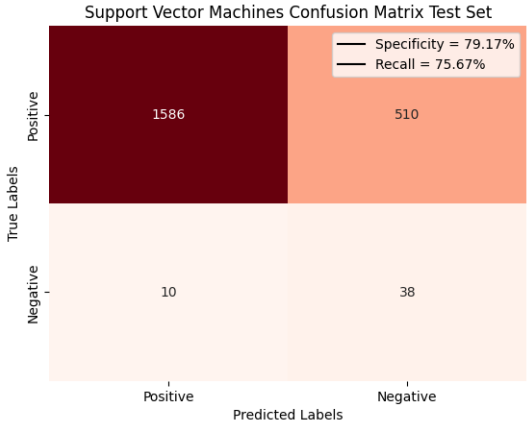
3.3.6 Naive Bayes



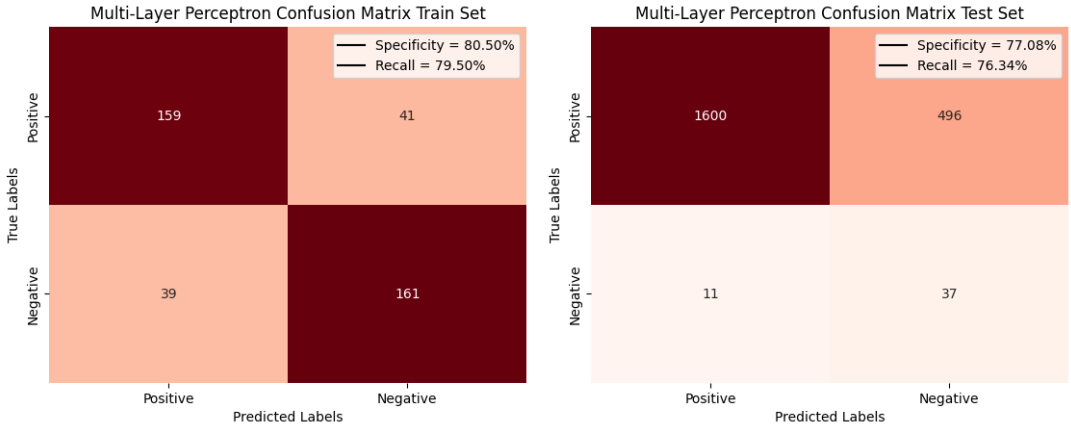


3.3.7 Support Vector Machines





3.3.8 Multi-Layer Perceptron



3.3.9 Overall Results

Classifier Name	Set type	Number of training samples	Number of non-healthy companies in training sample	TP	TN	FP	FN	Precision	Recall	Specificity	F1-Score	Accuracy
Linear Discriminant Analysis	Train	400	200	164	151	49	36	0.77	0.82	0.755	0.794	0.788
Linear Discriminant Analysis	Test	400	200	1618	37	11	478	0.993	0.772	0.771	0.869	0.772
Logistic Regression	Train	400	200	159	156	44	41	0.783	0.795	0.78	0.789	0.788
Logistic Regression	Test	400	200	1620	34	14	476	0.991	0.773	0.708	0.869	0.771
Decision Tree	Train	400	200	200	200	0	0	1	1	1	1	1
Decision Tree	Test	400	200	1469	32	16	627	0.989	0.701	0.667	0.82	0.7
K-Nearest Neighbors	Train	400	200	164	168	32	36	0.837	0.82	0.84	0.828	0.83
K-Nearest Neighbors	Test	400	200	1539	35	13	557	0.992	0.734	0.729	0.844	0.734
Naive Bayes	Train	400	200	160	149	51	40	0.758	0.8	0.745	0.779	0.772
Naive Bayes	Test	400	200	1662	34	14	434	0.992	0.793	0.708	0.881	0.791
Support Vector Machines	Train	400	200	162	165	35	38	0.822	0.81	0.825	0.816	0.818
Support Vector Machines	Test	400	200	1586	38	10	510	0.994	0.757	0.792	0.859	0.757
Multi-Layer Perceptron	Train	400	200	160	160	40	40	0.8	0.8	0.8	0.8	0.8
Multi-Layer Perceptron	Test	400	200	1598	36	12	498	0.993	0.762	0.75	0.862	0.762

It is observed that, with this balancing of the training data, all models satisfy the success rates of the constraints.

4 Conclusions

Based on the results above, it is observed that it is necessary for the training set of the companies to be balanced with a ratio of classes 1 to 1 so that the classes are considered of equal importance by the respective model. Since it is more important to successfully predict a company that is going to go bankrupt, the model that will output the highest specificity is sought. The model that performs best on this metric and trained on the 1-to-1 training set is Support Vector Machines with a success rate of predicting companies that will bankrupt 79.17% and a success rate of predicting companies that will not bankrupt 75.67

For further improvement, more data could be collected from bankrupt companies so that training can be done on a larger data set which is balanced at the same time.