Predicting Firm Bankruptcy with Advanced Machine Learning

PURDUE UNIVERSITY

Introduction and Overview

Roshan Raj Singh as Lone Analyst

Project Summary:

- Develop a predictive model using econometric measures.
- Aim: To predict the likelihood of a firm filing for bankruptcy.
- Real-world data competition applying data mining algorithms.
- Challenge: Analyzing complex econometric data for financial insights.

Data and Evaluation Criteria

Data Preprocessing and Evaluation in Bankruptcy Prediction

Data Overview:

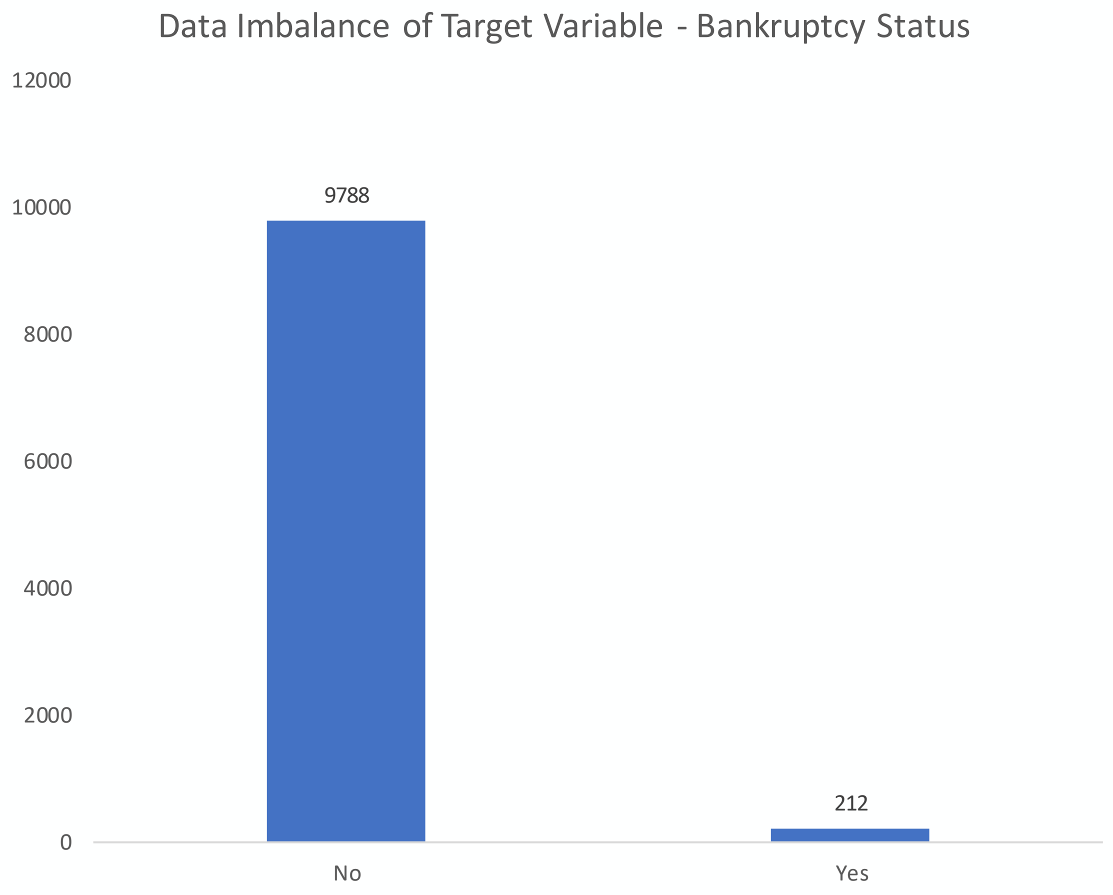
- Dataset: 10,000 records.
- Target Variable: Bankruptcy status.

Preprocessing Challenges:

- Missing and Outliers Identification.
- Imbalanced data: 212 out of 10,000 records as bankrupt.

Preprocessing Steps:

- Outlier Values: Capping Techniques.
- Balancing Data: Methods like over-sampling, under-sampling, SMOTE or choosing a robust ML algorithm which can overcome this issue.



PURDUE UNIVERSITY

Model Development - Outliers and Feature Selection

Building the Predictive Model: Refining Data and Features

Outlier Detection and Handling:

- Variables with kurtosis index >20, values were capped using boxplot whiskers.
- Remaining variables capped within 2 standard deviations for outlier mitigation.

Variable	Role	Mean	Deviation	Missing	Missing	Minimum	Median	Maximum	Skewness	Kurtosis	Imputed_Min	Imputed_Max
Attr38	INPUT	0.007436	0.019846	6999	0	-0.32493	0.00731	0.342475	0.944335	172.8809	0.00536	0.00921
Attr10	INPUT	-0.01173	0.078961	6999	0	-2.43711	-0.01769	3.03142	21.13834	983.2919	-0.03262	-0.00007
Attr59	INPUT	-0.01674	0.144769	6999	0	-0.0354	-0.03377	3.226273	17.88626	367.2286	-0.03475	-0.02593
Attr44	INPUT	-0.01515	0.027925	6999	0	-0.03522	-0.0198	0.522124	13.36335	231.0745	-0.0321	0.00221
Attr1	INPUT	0.017013	0.116526	6999	0	-2.80784	0.02458	1.33002	-12.544	286.3144	-0.05877	0.11022
Attr43	INPUT	-0.01009	0.09728	6999	0	-2.24803	-0.01868	2.25438	5.465136	362.486	-0.0666	0.03455
Attr62	INPUT	0.002622	0.127261	6999	0	-3.29639	-0.00346	3.280653	-5.87199	275.3797	-0.17028	0.17584
Attr41	INPUT	0.013509	0.120316	6999	0	-3.07093	0.01679	1.40475	-14.0934	341.2335	-0.05876	0.10313
Attr54	INPUT	-0.0112	0.005098	6999	0	-0.14289	-0.01125	0.05014	-12.4206	307.8163	-0.01553	-0.00655
Attr28	INPUT	0.013954	0.122896	6999	0	-3.0299	0.01664	1.48845	-12.6884	302.6564	-0.0702	0.11191
Attr53	INPUT	-0.01927	0.051038	6999	0	-0.07011	-0.02636	1.808067	24.46598	795.2263	-0.04386	0.01492
Attr25	INPUT	-0.01503	0.041899	6999	0	-0.42433	-0.02038	0.399854	2.173211	62.95233	-0.03355	-0.00194
Attr20	INPUT	0.005562	0.016037	6999	0	-0.18773	0.00429	0.199577	4.176575	107.7015	0.00167	0.00744

Feature Selection and Model Setup:

- Employed backward elimination to select features, resulting in a refined set of 49 variables.
- Initial models were built using these selected features, incorporating necessary preprocessing steps like normalization.

Variables selected		
Attr1	Attr35	Attr61
Attr11	Attr36	Attr62
Attr12	Attr37	Attr63
Attr13	Attr38	Attr8
Attr14	Attr39	Attr9
Attr17	Attr4	
Attr19	Attr40	
Attr2	Attr41	
Attr22	Attr42	
Attr23	Attr45	
Attr24	Attr46	
Attr25	Attr47	
Attr26	Attr48	
Attr27	Attr49	
Attr28	Attr50	
Attr29	Attr51	
Attr3	Attr52	
Attr30	Attr55	
Attr31	Attr56	
Attr32	Attr58	
Attr33	Attr6	
Attr34	Attr60	

PURDUE UNIVERSITY

Model Selection and Comparison

Navigating Through Models to Predict Bankruptcy

Model Exploration and Comparison:

- Ran 11 different algorithms including Decision Tree, Logistic Regression, and Gradient Boosting.
- Models compared using average squared error and ROC index for performance evaluation.

Final Model Selection:

- Chose an Ensemble model combining Gradient Boosting and Neural Network.
- Selection was based on improved validation metrics after hyperparameter tuning.

Selected Model	Model Node	Model Description	Valid: Roc Index	Train: Average Squared Error	Train: Misclassification Rate	Valid: Average Squared Error	Valid: Misclassification Rate
Y	Boost	Gradient Boosting (3)	0.915	0.006405	0.008858	0.016014	0.018660
	Ensmbl	Ensemble (2)	0.908	0.013430	0.019288	0.017915	0.021659
	HPDMForest	Bagging_Random_Forest	0.888	0.016259	0.021146	0.018526	0.021326
	Reg	Backward_Logistic_Regression	0.874	0.014996	0.017860	0.018905	0.020993
	HPDMForest2	Random_Forest	0.874	0.017729	0.021146	0.019507	0.021326
	Neural	Neural Network (3)	0.872	0.015704	0.018145	0.019328	0.020993
	Reg2	Forward_Logistic_Regression	0.853	0.018509	0.020860	0.020556	0.022659
	Reg3	Stepwise_Logistic_Regression	0.850	0.018753	0.020860	0.020564	0.021993
	Tree	Decision Tree	0.759	0.018226	0.019574	0.019347	0.020327
	LARS	LASSO	0.729	0.020897	0.021860	0.020755	0.021659
	LARS2	LASSO	0.500	0.020699	0.021146	0.020871	0.021326

PURDUE UNIVERSITY

Hyperparameter Tuning and Model Performance

First Model Approach :

Gradient Boosting Parameters:

- Iterations: 1,000
- Seed: 12345
- Shrinkage: 0.01
- Train Proportion: 90%

Neural Network Parameters:

- Number of Hidden Units: 32
- Randomization Distribution: Normal
- Input Standardization: Standard deviation
- Activation Function: Logistic

Performance:

- ROC Index for the Ensemble Model: 0.923

Fit Statistics
Model Selection based on Valid: Roc Index (_VAUR_)

Selected Model	Model Node	Model Description	Valid: Roc Index	Train: Average Squared Error	Train: Misclassification Rate	Valid: Average Squared Error	Valid: Misclassification Rate
Y	Boost2	Gradient Boosting	0.931	0.00429	0.005286	0.01593	0.017994
	Ensmbl2	Ensemble	0.923	0.11637	0.006001	0.12227	0.019327
	Neural2	Neural Network	0.532	0.44606	0.021146	0.44604	0.021326

Second Model Approach - Simplified Process:

- Bypassed initial data preprocessing.
- Gradient Boosting with 200 trees and 0.05 shrinkage.
- Neural Network with default parameter settings.
- Achieved ROC Index: 0.946.

Fit Statistics
Model Selection based on Valid: Roc Index (_VAUR_)

Selected Model	Model Node	Model Description	Valid: Roc Index	Train: Average Squared Error	Train: Misclassification Rate	Valid: Average Squared Error	Valid: Misclassification Rate
Y	Ensmbl	Ensemble	0.946	0.009482	0.015576	0.015264	0.019987
	Boost	Gradient Boosting	0.931	0.002549	0.002572	0.013631	0.016989
	Neural	Neural Network	0.913	0.013177	0.014861	0.016127	0.017988
	HPNNA	HP Neural	0.868	0.019937	0.021578	0.021537	0.023318

PURDUE UNIVERSITY

Key Takeaways: Hyperparameter Tuning and Beyond

Impact of Hyperparameter Tuning:

- The critical role of tuning in achieving high validation results.
- Success with minimal preprocessing, highlighting the power of parameter optimization.

Efficiency in Model Building:

- Enhanced computational efficiency through effective tuning.
- The balance between computational resources and model performance.

Strategic Use of Time:

- Time saved from reduced computational burden reallocated to additional model building and analysis.

Future Directions:

- Interest in implementing oversampling or SMOTE to improve model performance.
- Addressing limitations in SAS EM and exploring more flexible tools or methods.

PURDUE UNIVERSITY®

Thank You!

QUESTIONS

Q&A

ANSWERS