Endogenous Prediction of Bankruptcy using a Support Vector Machine

Jorge Zazueta Gutiérrez^{a,b}

Andrea Chávez Heredia^c

Jorge Antonio Zazueta-Hernández^c

^a Corresponding author

^b *School of Economics, Universidad Autónoma de San Luis Potosí*

Av. Pintores s/n, Colonia Burócratas del Estado

San Luis Potosí, SLP, México, CP. 78213

(444) 174 4907

jorge.zazueta@uaslp.mx

^c *Department of Mathematics, Universidad de Guanajuato*

Camino del Jalisco 37A, Valenciana

Guanajuato, Gto., México, CP. 36240

jorge.zazuetahernandez@gmail.com

andrea.chavez@cimat.com

Endogenous Prediction of Bankruptcy using a Support Vector Machine

We build a global bankruptcy prediction model using a support vector machine trained only on firms' endogenous information in the form of financial ratios. The model is tested not only on entirely random unseen data but on samples taken from specific global regions and industries to test for prediction bias, achieving satisfactory prediction performance in all cases. While support vector machines are not easily interpretable, we explore variable importance and find it consistent with economic intuition.

Keywords: bankruptcy; support vector machines; global model

JEL: C

Introduction

Several attempts have been made to create bankruptcy prediction models with a good measure of success. Accuracy ranges from 73% to 99% (Talha Mahboob Alam, 2020). However, most of these models were based on data from a specific country or region. (David Alaminos, 2016) proposed a global model using logistic regression with data from three global regions (America, Europe, and Asia), reporting a test accuracy of 84.86% for his global model and 90.11% after adding dummy variables to account for regions. This approach is sensible given that the available feature space might not carry complete information, so the region variable acts as a control. However, by doing so, the model becomes, again, geography dependent.

An ideal global model should be independent of geography and rely solely on the company's endogenous information. Due to globalization, it is reasonable to expect a somewhat homogeneous global environment and financial practices, at least for publicly traded companies, making a feasible and valuable endogenous model. A characteristic of the models in the literature is the diversity in the predictive variables used, calling for a theoretical foundation of bankruptcy as an economic phenomenon. Endogenous model development informs theory by focusing on the firm's behaviour rather than its environment.

In this paper, we build a prediction model that uses only internal data in the form of financial ratios and test its performance in specific regions and industries, aiming for high out-of-sample prediction accuracy. We are interested not only in observed accuracy on a general random test set but also in the same region or industry's subsamples. To do this, we rely on the same dataset used by (David Alaminos, 2016), containing information of 468 companies in three continents.

Support Vector Machines

SVMs are highly flexible and robust classifiers. Although generally regarded as hard to interpret, variable importance can be measured by running ROC curve analysis on each predictor (Kuhn & Johnson, 2016). SVMs were first introduced in the computer science field by (Vapnik, 1996) during the 1990s at AT&T Bell Laboratories. Generally speaking, the SVM can be thought of as a linear classifier in a high-dimensional feature space that is non-linearly related to input space and can thus be applied to linear and non-linear classification and regression problems.

The underlying idea behind the algorithm is that, given a training data set where every point belongs to one of two categories, we want to create a hyperplane that separates the data set, maximizing the width of the gap between both categories (See Figure 1). This is: in a 2-dimensional space, this would be that we want to be able to draw a line that clearly separates the data into two different sections. The hyperplane equation can classify new examples by assuming the same distribution as the original training, input, and data sets. When we can find a hyperplane that completely classifies the points in a data set, we say that the data is *linearly separable*. However, in many practical problems, finding such a hyperplane might not be feasible or possible. To deal with non-linearly separable sets (See Figure 2), we introduce the *kernel trick*, which implicitly maps the data into a *feature space* of higher dimension.

Figure 1. A 1-hyperplane classifying two types of data in a 2-dimensional space.

Figure 2. A data set that a hyperplane cannot classify.

Maximal Margin Classifier and Support Vector Classifiers

Given the data set $\mathcal{D} = \{(\mathbf{x}_i, y_i) \mid \mathbf{x}_i \in \mathbb{R}^p \text{ and } y_i \in \{1, -1\}\}_{i=1}^n$, our objective is to know if we can separate its points with a $(p - 1)$ -dimensional hyperplane. This is what we call a *linear classifier*. Among the multiple hyperplanes that may classify the data, a natural choice is the separating hyperplane that is farthest from the training observations; this natural choice is known as the *maximal margin hyperplane* or optimal separating hyperplane. We can then classify a test observation based on which side of the maximal margin hyperplane it lies. This is known as a *maximal margin classifier*.

In practice, linearly separable sets are rare. *Support vector classifiers (SVCs)* generalize the maximal margin classifier by allowing some points to be incorrectly classified, on the wrong side of the margin (See Figure 3). This is achieved by introducing slack variables in the underlying optimization problem. For this reason, SVCs are sometimes referred to as *soft margin classifiers*. By relaxing perfect separation, we obtain a more robust model that typically performs better at classifying out-of-sample points. Note that the hyperplane depends only on those data points that lie on the margin, called *support vectors*, making the classifier robust to new data that lies outside the margin.

Figure 3. A Support Vector Classifier allows for some points to cross the margin.

Dealing with Non-linearity

When a linear classification boundary is appropriate, SVC is a natural approach. However, it is common in practice to come across instances in which a non-linear boundary is necessary, as illustrated in Figure 2. The *support vector machine (SVM)* is an extension of the support vector classifier that results from mapping the input data into a high dimensional feature space via a non-linear transformation in order to perform the linear algorithm in the enlarged space, resulting in a non-linear boundary when projected back onto the original space.

Without delving into the details of the calculation, finding the maximal margin hyperplane is a quadratic programming problem that depends only on inner products between points in the feature space, as does the final decision function:

$$f(\mathbf{x}) = \text{sign}\left(\sum_i \alpha_i (\mathbf{x} \cdot \mathbf{x}_i + b)\right) \quad (1)$$

As we are working on the enlarged feature space, we substitute the inner product in (1) by the function $K(\mathbf{x}, \mathbf{y}) = \varphi(\mathbf{x}) \cdot \varphi(\mathbf{y})$. Where φ is a (typically) non-linear function that maps the input space into the enlarged feature space. Due to the high dimensionality of the feature space, evaluating all of the inner products can be computationally challenging.

However, there are simple *kernels* that can be evaluated efficiently by performing operations directly on the input space.

In this paper, we will utilize a kernel commonly used by practitioners, the radial basis function (RBF) kernel:

$$K(\mathbf{x}, \mathbf{y}) = \exp(-\sigma \|\mathbf{x} - \mathbf{y}\|^2) \quad (2)$$

Our specific decision function takes the form:

$$f(\mathbf{x}) = \text{sign} \left(\sum_i \alpha_i (\exp(-\sigma \|\mathbf{x} - \mathbf{x}_i\|^2) + b) \right) \quad (3)$$

The parameters α_i and b are computed via the quadratic programming problem, and σ is a hyperparameter that needs to be tuned by the practitioner, along with a regularization cost hyperparameter C that is part of the optimization problem. From (3), we can see that when the Euclidean distance between a new observation and a training point becomes large, its corresponding component becomes very small, implying that the radial kernel has local behaviour in the sense that only nearby training observations affect the class label. A detailed discussion of the algorithm can be found in (Gareth James, 2017), (Kowalczyk, 2017) and (Trevor Hastie, 2009)

Software and Tools

All of our calculations are performed in R (Team, 2019), with a heavy reliance on the *caret* package (Kuhn M. , 2008). Tables were handled by flextable (David Gohel [aut, 2021) and Graphs were created using *ggplot2* (Wickham, 2016) and *Mathcha* (Nha, 2021).

Model Development

The Data

In his paper, (David Alaminos, 2016) shared a global dataset including data from 468 publicly traded companies in 22 countries across Asia, Europe, and America from the period 1990-2013. Features include country, financial ratios (See Table 7), and industry code. Two hundred thirty-four of the companies in the sample legally declared bankruptcy. The rest of the sample was selected randomly from active companies following a match criterion by country, industry, and year. Financial ratios for each company correspond to the year prior to bankruptcy. The distribution of companies in the sample is summarized in Table 1. We will rely on this data to build and test our model. The overall methodology is depicted in Figure 4.

Country	Asia	Europe	America	Total	Bankrupt
Japan	112			112	56
Korea	8			8	4
Singapore	2			2	1
Taiwan	2			2	1
Austria		6		6	3
Bermuda		2		2	1
Denmark		20		20	10
France		32		32	16
Germany		22		22	11
Ireland		2		2	1
Italy		8		8	4
Luxembourg		2		2	1
Netherlands		8		8	4
Norway		10		10	5
Poland		8		8	4

Portugal	2			2	1
Spain	2			2	1
Sweden	22			22	11
Switzerland	2			2	1
United Kingdom	26			26	13
United States of America		154		154	77
Canada		16		16	8
Total	124	174	170	468	234

Table 1. Company distribution by country and region.

Data Preparation

Our dataset has a total of 73 missing values, distributed by feature as of Table 2. We impute the missing values using Multivariate Imputation by Chain Equations as implemented by the *mice* package in R (Stef van Buuren, 2011). For numerical data, mice uses predictive mean matching to impute each incomplete variable with a separate model.

Ratio	Missing
Earnings/Total Assets	9
Current Assets/Current Liabilities	9
Working Capital/Total Assets	7
Retained Earnings/Total Assets	10
EBIT/Total Assets	6
Sales/Total Assets	2
(Current Assets+Cash Flow)/Current Liabilities	8
Total Debt/Total Assets	7
Current Assets/Total Assets	3
Earnings/Net Worth	12

Table 2. Number of missing data by feature.

From our imputed data, we generate a test set by randomly selecting 25% of the data and training a Support Vector Machine with a radial basis kernel with the remaining 75%. The training set is stripped from region and industry information to develop a model based solely on the ten financial ratios available and subsets of the test data by region and industry to assess model performance on specialized subsamples (*i.e.*, only Asian countries or Consumer Discretionary focused companies).

Figure 4. Model development methodology.

Model Fitting

To fit our SVM we use repeated 10-fold cross validation with ROC's AUC as a selection metric. For hyperparameter tuning we resort to the empirical observation that the optimal values of the parameter σ in RFG kernel lie between the .10 and .90 quantiles of the $\|\mathbf{x} - \mathbf{x}_i\|^2$ statistics (Caputo, Sim, & Smola, 2002). The *sigest* function from *kernlab* uses a sample to estimate the quantiles and returns a vector with the .10, .50 and .90 quantiles. We run a grid search with these three values and cost, C , ranging from 2 to 200 in steps of 0.5. with scaled and centered data. Our results are shown in

Figure 5. With the best model's hyperparameters being $C = 9.5$ and $\sigma = 0.1497$.

Figure 5. Cost hyperparameter tuning.

Results

Performance Metrics

The final model has an average cross validated accuracy of 0.842 and a percentual average confusion matrix shown in Table 3.

		Reference	
		Yes	No
Prediction	Yes	39.69%	5.48%
	No	10.31%	44.52%

Table 3. Final Model Cross Validated Confusion Matrix

Overall prediction performance on the test set is summarized in

Table 4. Where the first four reported performance indicators are defined in terms of the confusion matrix elements as follows:

$$Accuracy = \frac{\text{number of true positives} + \text{number of true negatives}}{\text{number of positives} + \text{number of negatives}}, \quad (4)$$

$$Sensitivity = \frac{\text{number of true positives}}{\text{number of true positive} + \text{false negatives}}, \quad (5)$$

$$Specificity = \frac{\text{number of true negatives}}{\text{number of true negatives} + \text{false positives}}, \quad (6)$$

$$Kappa = \frac{Accuracy - Expected Accuracy}{1 - Expected Accuracy}, \quad (7)$$

And ROC is the area under the Receiver Operator Characteristic curve. Performance on the overall testing set is quite satisfactory, with accuracy of 0.88 and an ROC of 0.93.

n	Accuracy	Sensitivity	Specificity	Kappa	ROC
116	0.888	0.879	0.897	0.776	0.934

Table 4. Overall, out-of-sample model performance

The model also performs well on specific region, with similar performance.

Region	n	Accuracy	Sensitivity	Specificity	Kappa	ROC
Asia	34	0.853	0.875	0.833	0.706	0.924
America	41	0.878	0.857	0.900	0.756	0.902
Europe	41	0.927	0.905	0.950	0.854	0.957

Table 5. Model performance metrics by region

From the training sample, we focused on industry subsets with more than ten elements, recovering *Industrials*, *Consumer Discretionary* and *Information Technology*. Model performance remains stable with similar scores.

Industry	n	Accuracy	Sensitivity	Specificity	Kappa	ROC
Industrials	28	0.893	0.933	0.846	0.784	0.974
Consumer Discretionary	40	0.900	0.857	0.947	0.800	0.937
Information Technology	26	0.885	0.933	0.818	0.761	0.927

Table 6. Model performance by industry.

Interpretability

Feature importance was calculated with the caret package and is depicted in Figure 6. It is interesting to note that, according to the model, the most relevant variables belong to the profitability and liquidity categories, followed closely by debt. It is also noteworthy that efficiency, as represented by *sales/(total assets)* doesn't play a significant role in bankruptcy prediction. This structure is reasonable from an economic and financial standpoint,

suggesting that a purposeful exploration of larger feature sets might shed some light on a theoretical formulation of the bankruptcy phenomena.

Figure 6. Feature importance by ratio type.

Study Limitations and Future Research

As any numerical exploration, our results are limited by the characteristics of the sample. A serious limitation is the availability of features. Replicating this exercise on a feature-rich sample will help us better understand the relationship between endogenous company information and bankruptcy. Our modelling effort was limited—by design—to one period in advance prediction. Developing earlier warnings is a topic of interest in the literature that might be improved by incorporating time series-based models as well as a theoretical foundation.

References

Boser, B. G. (1992). A training algorithm for optimal margin classifiers. *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*.

- Caputo, B., Sim, K., & Smola, A. (2002). Appearance-based Object Recognition using. *Proceedings of NIPS Workshop on Statistical Methods for Computational Experiments in Visual Processing and Computer Vision*. Whistler.
- Cortez, P. (2010). Data Mining with Neural Networks and Support Vector Machines using the R/rminer Tool. In P. Perner, *Advances in Data Mining -- Applications and Theoretical Aspects, 10th Industrial Conference on Data Mining* (pp. 572--583). Berlin: Springer.
- David Alaminos, A. d. (2016). A Global Model for Bankruptcy Prediction. *PLoS ONE*.
- David Gohel [aut, c. C. (2021, April 6). *flextable: Functions for Tabular Reporting*. Retrieved from cran.r-project.org: <https://cran.r-project.org/web/packages/flextable/>
- Gareth James, D. W. (2017). *An Introduction to Statistical Learning*. New York, Heidelberg, Dordrecht, London: Springer.
- Kowalczyk, A. (2017). *Support Vector Machines Succinctly*. Morrisville, NC: Syncfusion.
- Kuhn, M. (2008). Building Predictive Models in R Using the caret Package. *Journal of Statistical Software*.
- Kuhn, M., & Johnson, K. (2016). *Applied Predictive Modeling*. New York: Springer.
- Nha, B. D. (2021, April 6). Retrieved from Mathcha: <https://www.mathcha.io/>
- Pablo Cortez, , J. (2013). Using sensitivity analysis and visualization techniques to open black. *Information Sciences*, 1-17.
- Stef van Buuren, K. G.-O. (2011). mice: Multivariate Imputation by Chained Equations in R. *Journal of Statistical Software*.
- Talha Mahboob Alam, K. S. (2020). Corporate Bankruptcy Prediction: An Approach Towards Better Corporate World. *The Computer Journal*.

- Team, R. C. (2019). *R: A Language and Environment for Statistical Computing*. Viena: R Foundation for Statistical Computing.
- Trevor Hastie, R. T. (2009). *The Elements of Statistical Learning*. New York: Springer-Verlag.
- van Buuren, S., & Groothuis-Oudshoorn, K. (2011). mice: Multivariate Imputation by Chained Equations in R. *Journal of Statistical Software*.
- Vapnik, V. (1996). *The Nature of Statistical Learning Theory*. New York: Springer.
- Wickham, H. (2016). *ggplot2: Elegant Graphics for Data Analysis*. New York: Springer-Verlag.

Appendix

Ratio	Type
Earnings/Total Assets	Profitability
Current Assets/Current Liabilities	Liquidity
Working Capital/Total Assets	Liquidity
Retained Earnings/Total Assets	Profitability
EBIT/Total Assets	Profitability
Sales/Total Assets	Efficiency
(Current Assets + Cash Flow)/Current Liabilities	Liquidity
Total Debt/Total Assets	Leverage
Current Assets/Total Assets	Liquidity

Earnings/Net Worth

Profitability

Table 7. Financial ratios used to train the model and their classification, consistent with Alaminos' paper.