Utilizing principal component analysis to enhance machine learning in bankruptcy prediction: A comparative investigation

Xinming Ma

University of Minnesota, twin cities, College of Liberal Arts, Minneapolis, Minnesota, The United States, 55414

ma000572@umn.edu

Abstract. In the global economic environment, bankruptcy prediction is essential for managing financial risks. With the advancement of machine learning techniques, accurate prediction using these methods has become a research focus. This study explores the efficiency and accuracy of improved machine learning models for bankruptcy prediction through principal component analysis (PCA). By utilizing a dataset from the banking industry in Taiwan, this paper compares the performance of PCA with and without processed data using logistic regression modeling. The research methodology includes data preprocessing, PCA downscaling, and subsequent model training and testing. The key research question is whether PCA preprocessing can significantly improve the operational efficiency and predictive accuracy of the model. It is found that the model with PCA outperforms the model without PCA in terms of processing time and accuracy. This suggests that PCA can effectively improve the performance of bankruptcy prediction models and provide a more effective tool for financial risk management. These findings provide useful insights into other areas of financial analysis using machine learning and support the value of applying PCA in predictive modeling.

Keywords: PCA, Finance, Logistic regression modeling, Corporate bankruptcy.

1. Introduction

In a globalized economy, accurate prediction of corporate bankruptcy is extremely critical for investors, creditors and policymakers. With the rapid development of data science and machine learning technologies, bankruptcy prediction models have shifted from traditional statistical methods to sophisticated algorithms that can efficiently handle large-scale datasets to improve prediction accuracy. This study aims to investigate the efficiency and accuracy of principal component analysis (PCA) in improving machine learning models for processing Taiwan banking datasets. By comparing the model performance of PCA preprocessed and unprocessed data, this paper will analyze the effectiveness of PCA in reducing model training time and improving prediction accuracy. The research questions focus on whether PCA can significantly improve the operational efficiency and predictive accuracy of models and explore the specific impact of this data preprocessing technique on model performance. The results of this study will not only help validate the practical application of PCA in financial risk management.

At the same time, they may also provide valuable insights into using machine learning for broader financial analysis.

2. Literature review

2.1. PCA method introduction

Principal Component Analysis (PCA) is a statistical method for downscaling high-dimensional data to a lower dimensional space by linear transformation while preserving as much variance information as possible in the data. It uses the eigenvalue decomposition of the covariance matrix to extract the principal components so that the new variables after dimensionality reduction are linear combinations of the original variables and that these new variables are uncorrelated [1].

2.2. Logistic Regression introduction

Logistic regression is a statistical model for binary classification problems that predicts the probability of an input sample belonging to a particular category by learning the relationship between input features and output labels. It transforms linear combinations of input features into probabilities using a Sigmoid function and optimizes the model parameters by minimizing a log-loss function [2].

For Sigmoid function, we use this:

$$\sigma(z) = \frac{1}{1+e^{-z}} \tag{1}$$

2.3. PCA in machine learning

Principal Component Analysis (PCA) in machine learning reduces the complexity of the data by reducing dimensionality, removing noise and redundant information, and thus improving the training efficiency and performance of the model.PCA helps in feature selection, reduces the risk of overfitting, improves data visualization, and enhances the performance of algorithms on high-dimensional data. Its core steps include data centering, calculation of covariance matrix, eigenvalue decomposition and data projection into principal component space [3].

2.4. A comparative study of model efficiency and accuracy

For this research, we use the F1 score to analysis the efficiency and accuracy of the model.

F1 Score is a metric that combines Precision and Recall to evaluate the performance of a classification model, balancing the accuracy and coverage of the model by their reconciled averages, and is particularly suitable for datasets with imbalanced categories, ensuring that the model's ability to recognize a small number of classes is not overlooked [4]. The formula is below

$$\mathrm{F1} = 2 \times \frac{\mathrm{Precision} \times \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}} \tag{2}$$

3. Characterization and processing of data

3.1. data description

The Taiwan Bankruptcy Prediction dataset in the UCI Machine Learning Repository contains financial data used to predict the bankruptcy status of Taiwanese companies. It consists of a feature matrix X containing various financial indicators and ratios and a target vector y indicating the bankruptcy outcome.

3.2. Data preprocessing

3.2.1. Data loading and initial exploration. First, the Taiwan company bankruptcy prediction dataset is loaded from the UCI machine learning library using the fetch_ucirepo function. The loaded dataset is divided into a feature matrix X and a target vector y. Next, the shape of the dataset is examined to understand its dimensions, including samples and the number of features.

3.2.2. Variance Visualization. After data loading and initial exploration, the variance of each feature in the dataset is calculated. By plotting the variance of each feature, it is possible to identify features with low variance that may contribute less to the predictive power of the model.

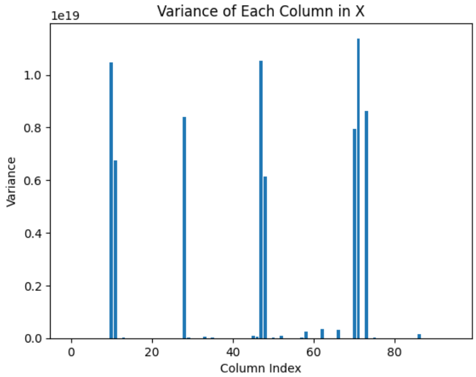

Figure 1. Variance of Each column in X

This figure shows the distribution of variance for each feature in the dataset, revealing that some features have very high variance, while most features have variance close to zero. Features with high variance are clustered around specific column indexes, indicating that these features have large variations in the data.

3.2.3. data segmentation. For model training and evaluation, the dataset is partitioned into training and test sets using a 70%:30% ratio. The train_test_split function is used to allocate 70% of the data for training the model and 30% for testing it. This ensures that the model learns patterns from the training data and is evaluated on unseen test data to assess its generalization ability. This split helps measure the model's actual performance and avoid overfitting. Using appropriate random seeds ensures reproducibility and facilitates result validation and comparison.

3.2.4. Feature standardization. First, the features are normalized using StandardScaler to ensure that all features have a mean of 0 and a standard deviation of 1. This step is important for many machine learning algorithms because it removes the effects between different feature scales and allows the model to learn patterns in the data better. PCA is then applied before and after standardization, and by visualizing the results of PCA, the effect of standardization on the distribution of the data can be visualized.

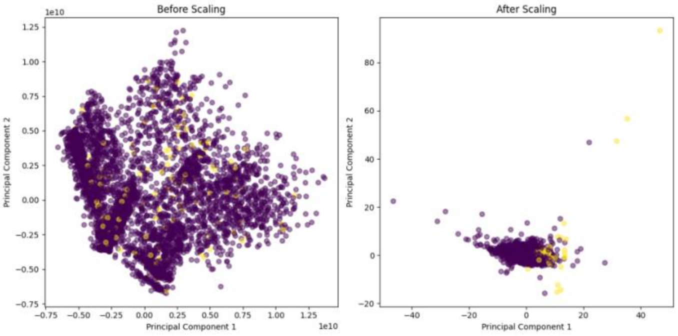

Figure 2. Applied PCA before and after standardization

Figure 2 illustrates the impact of standardization on principal component analysis (PCA) results. The left plot, showing PCA before standardization, displays a dispersed distribution of data points with no clear structure due to the scale differences between features. The right plot, showing PCA after standardization, reveals a more concentrated and compact data distribution. This indicates that standardization removes feature scale differences, enabling PCA to more effectively capture the main variation patterns in the data, thus making the data more interpretable and structured in the reduced space.

3.2.5. Category imbalance check. In order to identify any category imbalance problem, it is first necessary to examine and visualize the distribution of the target variable (y). As category imbalance can significantly affect the performance of a model, causing it to favor the majority class and ignore the minority class.

The distribution of categories for the target variable (y) reveals a significant category imbalance. Category 0 (non-bankrupt firms) has about 6,600 samples, while Category 1 (bankrupt firms) has only about 220 samples. This imbalance leads to a tendency for the classification model to predict the majority class (class 0) and ignore the minority class (class 1), thus affecting the predictive performance of the model. To cope with this imbalance, the F1 score should be used as an indicator when evaluating model performance.

3.2.6. feature filter. In data preprocessing, to reduce the dimensionality of the dataset and remove features that may not be informative, we need to identify and remove features with only two unique values (0 and 1) [5]. By removing these low-information features, we can simplify the dataset and reduce the computational complexity, thus improving the efficiency and performance of model training [6].

3.3. Application of PCA

3.3.1. Cumulative Variance Plot. First, PCA is performed on the normalized training data to project the data into a lower dimensional principal component space. This process preserves the maximum amount of information in the data while reducing the number of features. Next, cumulative variance plots of principal component explanations are plotted to determine how well each principal component explains the variance of the data [7].

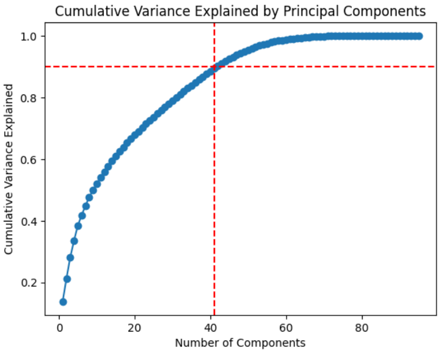

Figure 3. Cumulative variance explained by Principal components

Figure 3 illustrates the cumulative proportion of variance explained by each principal component in PCA. The cumulative variance explained increases with the number of principal components and levels off after a certain point. The figure shows that about 40 principal components explain over 90% of the variance. This indicates that selecting the first 40 principal components significantly reduces data dimensionality while retaining most information.

3.3.2. PCA processing. We performed the PCA fitting process outside the function to ensure the purity of the input data each time the function was tested and to prevent the PCA fitting process from affecting the time statistics. By placing the PCA fitting process outside the function, we ensure that the PCA transformation is performed based on the entire training data, avoiding the temporal bias introduced by re-fitting the PCA at different function calls.

3.4. function definition

To evaluate the impact of using raw normalized features versus PCA-transformed features on the performance of the logistic regression model, two functions were defined: process_without_pca and process_with_pca. The process_without_pca function uses normalized raw features for training and evaluation, recording the F1 score and total runtime. The process_with_pca function performs the same process using PCA-transformed features, ensuring PCA fitting is done outside the function. By comparing the F1 scores and runtimes of these two functions, the impact of PCA on feature degradation and model performance can be systematically assessed to determine the optimal data preprocessing and modeling strategy.

4. Repeat the test and compare

4.1. Repeat the test procedure

In order to assess the difference in the performance of the logistic regression model when using the original normalized features and the PCA transformed features, we performed a repeatability test for both functions and compared the results [8]. Specifically, we performed between 10 and 50 repetitions of the experiment, recording the model's F1 score and average running time, and representing these results in line graphs. In addition, we calculated the total running time for repetitions ranging from 10 to 50 and plotted the results on a graph.

In each repetition of the experiment, the logistic regression model was trained and evaluated using the original normalized features and the PCA transformed features, respectively. Each experiment’s start and end times were recorded to compute the running time and predicted on the test set to obtain F1 scores. These results were averaged in order to compare the performance of the models under different feature treatments.

By plotting a line graph of the F1 score and the average run time, it is possible to observe how the model's performance varies over different numbers of repetitions. The difference in computational efficiency between different feature processing methods is then demonstrated by comparing the histograms of the total running time.

4.2. Results Charts and Interpretation

The F1 scores of logistic regression models with and without PCA were measured over 10 to 50 repetitions. Both models showed stable F1 scores: around 0.27 with PCA and 0.34 without PCA. This indicates that the model without PCA outperforms the model with PCA. The stability of the F1 scores also suggests consistent performance regardless of the number of repetitions. Overall, the logistic regression model using standardized features performs better than the model using PCA-transformed features.

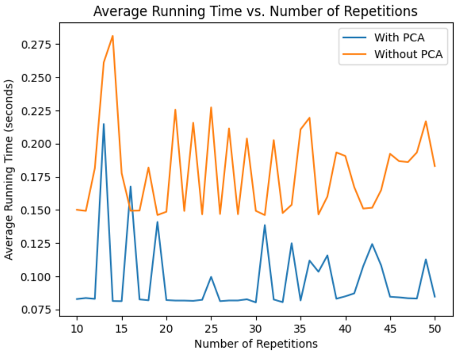

Figure 4. Average running time comparisons

Figure 4 shows how the average running time of the logistic regression model varies with the number of repetitions for both methods, with and without PCA, when 10 to 50 repetitions of the experiment are performed. The horizontal axis indicates the number of repetitions and the vertical axis indicates the average running time in seconds. The blue line indicates the model using PCA and the orange line indicates the model without PCA.

As can be seen from the figure, the average run time of the model using PCA is low, fluctuating between about 0.075 and 0.15 seconds. Meanwhile, the models without PCA have longer run times, fluctuating between 0.15 and 0.275 seconds. Overall, the model using PCA was more stable and faster in terms of run time, and despite a brief spike at some repetitions, the overall run time was still significantly lower than the model without PCA.

This suggests that under the current dataset and model configuration, PCA significantly improves the efficiency of model training and prediction while reducing the data dimensionality. Although the previous F1 score results show slightly worse model performance using PCA, the advantage in computational efficiency makes the PCA transformation an effective option for dealing with large-scale data or when computational resources are limited.

To assess the performance difference in the logistic regression model using original normalized features versus PCA-transformed features, we conducted repeatability tests with 10 to 50 repetitions. We recorded the F1 score and average running time for each method.

In each repetition, the logistic regression model was trained and evaluated using both feature sets. We recorded the start and end times to compute running times and predicted on the test set to obtain F1 scores. These results were averaged to compare the performance of the models.

By analyzing the F1 scores and average run times, we observed how the model's performance varied with the number of repetitions. The computational efficiency differences between the two feature processing methods were evident from the total running times. The method using PCA had a total running time of about 4 seconds, whereas the method without PCA had a total running time of nearly 7.5 seconds, indicating that the PCA method is faster.

5. Conclusion

In summary, by testing and comparing the logistic regression model using the original normalized features and the PCA transformed features over multiple repetitions, we find that although the model using PCA has slightly lower F1 scores than the model not using PCA, it has a significant advantage in terms of running time. Specifically, the PCA approach exhibits lower average and total running time in multiple repetitive experiments, significantly improving computational efficiency. This suggests that in the case of high data dimensionality or limited computational resources, PCA, as a dimensionality reduction technique, not only effectively reduces the data dimensionality, but also improves the efficiency of model training and prediction, albeit with a slight performance loss. Overall, PCA provides an effective solution for large-scale data processing, balancing computational efficiency and model performance well.

References

[1] Sanguansat, P. (2012). Principal Component Analysis (P. Sanguansat, Ed.). IntechOpen.

[2] Cornilly, D., Tubex, L., Van Aelst, S., & Verdonck, T. (2023). Robust and sparse logistic regression. Advances in Data Analysis and Classification. https://doi.org/10.1007/s11634-023-00572-4

[3] Sun, Ping'an, & Wang, Bizhan. (2019). Research on PCA dimensionality reduction method in machine learning and its application. Journal of Hunan University of Technology, 33(1), 73–78. https://doi.org/10.3969/j.issn.1673-9833.2019.01.012

[4] Chicco, D., & Jurman, G. (2020). The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. BMC Genomics, 21(1), 6–6. https://doi.org/10.1186/s12864-019-6413-7

[5] Jain, A. K., Duin, R. P. W., & Jianchang Mao. (2000). Statistical pattern recognition: a review. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(1), 4–37. https://doi.org/10.1109/34.824819

[6] Tan, P.-N., Steinbach, M., & Kumar, V. (2005). Introduction to data mining (1st ed.). Pearson Addison Wesley.

[7] Jackson, J. E. (1991). A user’s guide to principal components. Wiley.

[8] Goodman, S. N., Berlin, J., Fletcher, S. W., & Fletcher, R. H. (1994). Manuscript quality before and after peer review and editing at Annals of Internal Medicine. Annals of Internal Medicine, 121(1), 11–21. https://doi.org/10.7326/0003-4819-121-1-199407010-00003