
Research article

Prediction of IPO performance from prospectus using multinomial logistic regression, a machine learning model

Mazin Fahad Alahmadi and Mustafa Tahsin Yilmaz*

King Abdulaziz University, Engineering Faculty, Industrial Engineering Department, 21589, Jeddah, Saudi Arabia

Corresponding: E-mail: myilmaz@kau.edu.sa.

Abstract: In this study, we investigated and determined the financial indicators that significantly affect the performance of initial public offerings (IPOs) by using the Multi-Logistic Regression (MLR) method based on several financial variables as independent variables extracted from prospectuses. The prospectus serves as an important source of information for potential investors as it significantly increases the likelihood of attracting their attention. A total of twelve characteristics in two segments, namely “prospectus characteristics” and “financial ratios”, were used as the features to assess IPO performance in the Saudi stock market in three categories: BELOW AVERAGE, AVERAGE, and ABOVE AVERAGE performance. The accuracy, recall, precision, and F1 scores as well as the confusion matrix and AUC were used to evaluate our dataset. Python was used, as an Integrated Development Environment (IDE), to develop the model. Based on the results of the classification analysis, 71.4% of the predictions were accurate with an AUC of 0.71 for the ABOVE class. The most significant financial variable that affected IPO returns was the subscription quarter (SQ), followed by sector code (SC), and recent year net profit margin (NPM%). The MLR model had a higher level of accuracy when compared with other machine learning algorithms. By using the model developed here, investors can improve their ability to predict the direction of the return on their investment in an IPO, at least for the first month. A practical application of the MLR method is discussed in the paper, along with how it can be used to predict the probability of the performance of an IPO in the future.

Keywords: initial public offering (IPO) returns; machine learning algorithms; multinomial logistic regression; performance; python

JEL Codes: C25, C45, C53, G14, G32

1. Introduction

Companies often raise funds through their initial public offering (IPO). An initial price is set and a certain number of shares are offered to the public at these times (Ashford, 2023). Although raising new capital for growth is the most common reason for going public, other factors may also be at play. There are also other reasons for the preference: To lower the cost of capital, to facilitate takeovers, to offer an opportunity to insiders for financial gain, and to obtain a first-mover advantage (Emidi et al., 2022). In the first day of trading, the price may exceed, remain the same, or fall below the initial offering price. In the former case, the product is underpriced, e.g. below average, while in the latter case, the product is overpriced, e.g., above average (Kagan, 2023).

IPO return predictions have been challenging because a large number of determinants have different explanatory power and outliers influence returns. Therefore, prediction of the performance of IPO returns is a critical area of research in the field of finance and capital markets. There has been limited attention paid to predicting the outcome of IPOs compared with other measures of financial performance. However, there have been researchers who attempted to predict the outcome of an IPO; in this respect, there are some instances in which text data can be used to develop predictors. Prospectuses are one of the most frequently used text documents and investors believe that the prospectus contains all the necessary information about the company (Emidi et al., 2022; Hanbing et al., 2019). Prospectuses play a crucial role in predicting IPO outcomes as they contain detailed information about the company's financial health, business model, risks, and future prospects. By analyzing the information given in the prospectuses, researchers can uncover valuable insights that help in assessing the likelihood of a successful IPO and predicting its future performance. This information is particularly useful for investors and market analysts in making informed decisions and managing investment risks. However, previous IPO prediction methods have been limited in their ability to capture the full range of outcomes and complexities associated with stock market debuts. Many existing approaches rely on simplistic models or overlook important factors that can influence IPO performance. As a result, their predictive accuracy is often limited, leaving investors and researchers with an incomplete understanding of the dynamics at play. Our innovative use of multinomial logistic regression addresses these limitations by providing a more nuanced and comprehensive analysis that takes into account a wider range of variables and potential outcomes.

The groundbreaking nature of our paper lies in its innovative approach to forecasting Initial Public Offering (IPO) outcomes. While researchers have delved into IPO prediction, we pioneer the utilization of multinomial logistic regression as a predictive tool. This sophisticated statistical technique allows for a more nuanced analysis of IPO performance, capturing the inherent complexities and diverse outcomes associated with stock market debuts. Moreover, our study breaks new ground by scrutinizing prospectuses, an underexplored but rich source of information. The comprehensive examination of prospectus content, coupled with the application of multinomial logistic regression, represents a novel intersection of methodologies, promising a more accurate and insightful predictive model. By introducing this pioneering framework, our research not only advances the predictive capabilities in the field but also contributes a unique perspective that has the potential to reshape how

IPO performance is understood and anticipated in both academic and practical contexts. Therefore, our paper focuses on two segments, namely “prospectus characteristics” and “financial ratios” representing a company’s uniqueness. This means that we do not focus on the sentiments and connotations related to certain words or statements, but rather on these two segments. In this paper, we define these segments based on the information found in the prospectus of the Saudi Arabian capital market authority (CMA). The prospectus, therefore, is unique in that it contains a different information of the defined financial features. Therefore, we consider these information to be representative of a company’s idiosyncrasy. Therefore, our purpose of this paper is to contribute to the literature by investigating whether the uniqueness of the prospectuses, which is reflected in their topics, can be used to predict the returns of an initial public offering.

Our first step is to identify the topics represented by each document. The predictive phase of our study was based on a total of twelve features from the two segments. Our final step is to utilize these features in order to predict the class of an IPO. These outcomes can be categorized as BELOW AVERAGE (=0), AVERAGE (=1), or ABOVE AVERAGE (=2). Data is split into train and test data, and predictions are made using multinomial logistic regression (MLR). MLR is useful as a supervised machine learning algorithm and classification model for predicting the class of a multichotomous dependent variable. One advantage of using multinomial logistic regression (MLR) for classification is that it allows for the prediction of multiple classes simultaneously, which is ideal for situations where the dependent variable has more than two categories. Additionally, MLR provides probabilities for each class, giving a measure of uncertainty in the predictions. This information can be valuable for decision-making and understanding the reliability of the model’s classifications. Model performance was assessed by calculating the accuracy, recall, precision, and F1 scores, as well as the confusion matrix and AUC. To develop the model, Python was used as an Integrated Development Environment (IDE). Additionally, the accuracy of MLR was compared with that of other machine learning algorithms.

2. Literature review

In recent years, much research has been conducted concerning the prediction of financial performance from a variety of perspectives. Researchers have been particularly interested in stock market fluctuations as a measure of financial performance. Fundamental and technical analysis have both been used to address the challenge of predicting the development of stocks in the market. A third approach (Nguyen et al., 2015; Schumaker et al., 2009) is the analysis of text to identify significant features. In order to predict the movement of stock prices, various sources of texts have been utilized: Financial news (Schumaker et al., 2009; Schumaker et al., 2012), social media comments (Nguyen et al., 2015), financial disclosures (Kraus et al., 2017), and 10-K forms (Loughran et al., 2011). On the other hand, Tao et al. (2018) conducted a study in which they attempted to forecast IPO outcomes by referring to forward-looking statements in their prospectus. In order to accomplish this, they utilize a section of the prospectus, namely the Management’s Discussion and Analysis, which is regarded as one of the most informative sections. By means of their prospectuses, corporations convey information to investors that they elaborate in order to gain an understanding of the company as well as to assess the profitability of investing in such a company. Therefore, their willingness to purchase IPO shares will be affected (Daily et al., 2003). This has been demonstrated in many studies showing that critical information from the prospectuses play a significant role in the IPO process (Connelly et al., 2011).

Due to the limited availability of corporate information, investors rely on the prospectus and disclosure of information to determine the value of a business, especially quick start-ups. The only method through which investors can obtain information is through the public disclosure of the issuer's prospectus, and for further information, investors must pay additional charges. There has been a recent trend in mature capital markets in the West in which a company has issued new shares in its prospectus in anticipation of future earnings disclosures. It is of great importance to investors that management or analysts' earnings forecasts are based on what the firm expects to happen in the future. It is important to note that forecasts in the prospectus are intended to contribute to reducing the prevalence of information asymmetries in the IPO market (Hanbing et al., 2019).

While many studies have been conducted on predicting stock price movements through text data, there has been less research on predicting IPO outcomes (Emidi et al., 2022). One reason for the lack of research on predicting IPO outcomes is the inherent complexity and volatility of the IPO market. IPOs involve multiple factors such as market sentiment, investor behaviour, and regulatory changes, making it difficult to develop accurate and reliable prediction models. Additionally, IPO data is often limited and historical patterns may not be sufficient to make accurate predictions. Moreover, the lack of standardized data and the uniqueness of each IPO make it challenging to establish a comprehensive framework for prediction models. Furthermore, the dynamic nature of the IPO market and the influence of external factors, such as economic conditions and industry trends, add additional complexity to the prediction process. As a result, researchers and analysts face significant hurdles in accurately forecasting IPO outcomes.

Application of artificial intelligence to finance domains has become an increasingly hot topic. The performance of IPOs is the most important aspect of these domains (Ni, 2023). In this respect, several studies have been published on the use of multinomial logistic regression (MLR) to predict the outcome of initial public offerings. The studies have consistently found that logistic regression models can effectively predict the outcome of initial public offerings. They have identified several key variables, such as firm size, industry sector, and financial performance which significantly influence the likelihood of a successful IPO. These findings have important implications for investors and financial institutions involved in the IPO market. (Anand et al., 2023). Other researchers have examined the prediction of stock performance using a logistic regression model and evidence from the Pakistan Stock Exchange (PSX). The study analyzed various financial factors such as sales growth, debt to equity ratio, and earning per share (Ali et al., 2018). In another study, the use of random forest for predicting IPO initial returns was explored, highlighting the limitations of linear regression models in analyzing IPO initial returns (Baba et al., 2020). However, it was also reported that logistic regression performed well in sentiment categorization and stock market movement prediction (Das et al., 2022), providing valuable insights for capital market participants and researchers.

3. Methods

Research Objective: Our purpose of this study was to classify the performance of recently listed stocks based on their returns from their initial public offering (IPO) price. The classes in the categorical dependent variable are BELOW AVERAGE, AVERAGE, AND ABOVE AVERAGE. Their encoded values along with their definitions are shown in Table 1. We aimed to build a model using independent variables of the firms to predict the performance of outperforming shares on the Saudi Stock Market (Tadawul) with respect to their IPO offer prices. Hence, the following questions could be answered:

(a) Is it possible to explain the yields of these stocks using the independent variables? (b) Would a multi-logistic regression model be appropriate for analyzing stock yields? Moreover, we investigated the effectiveness of the independent variables selected as predictors of stock performance.

Table 1. Levels of dependent variable, encoded values along with their definitions.

Performance level	Encoded value	Definition
BELOW AVERAGE	0	Stocks that have negative returns within one month after listing (i.e., the stock price drops below its IPO offer price).
AVERAGE	1	Stocks that offer average returns within a month of listing (i.e., the stock price rises above its IPO offer price by between 1% and 59% within a month).
ABOVE AVERAGE	2	Stocks that provide an upnormal return within a month of listing (i.e., the stock price rises above the IPO offer price by greater than 60%).

Dataset: The data for IPO was collected manually from all PDF Prospectuses from the capital market authority (CMA) website (<https://cma.org.sa/en/Market/Prospectuses/Pages/default.aspx>). This is the official website of the Saudi Arabian Capital Market Authority (CMA), which is an excellent source for obtaining prospectuses of Saudi stocks. In the context of an IPO or any significant fundraising effort, a prospectus is an essential document that provides in-depth information about a company's finances and business model. By offering access to these documents on both the Main Market and the Nomu Parallel Markets, the CMA platform ensures transparency, enabling investors to make informed decisions.

In this paper, 68 companies that have gone public on the Saudi stock market between 2004 and 2023 were analyzed through carefully curated prospectus data. However, the final data set for the analysis contains only 55 companies, after excluding insurance firms due to the fact that their financial structures are one of a kind, which makes direct comparisons between them and other sectors ineffective. As a result, we empirically examined the performance for 1 month after the listing of 55 Saudi stock major exchange IPOs from 2010 to 2022. The data includes information on IPO prices, returns, and other relevant factors. The dataset is bifurcated into two key segments: Prospectus Characteristics and Financial Ratios. The former includes IPO-specific variables such as Sector Code (SC), Total Number of Offer Shares (TOS), Offer Price (OP), Number of Substantial Shareholders (SH), Subscription Quarter (SQ), Individual Coverage (ind_sub%), and Institutional Coverage (ins_cov%). The second segment provides a picture of the company's financial health immediately before the IPO year through financial ratios. These ratios include Net Profit Margin (NPM%), Return on Assets (ROA%), Current Assets to Current Liabilities (CR), Liabilities to Equity Ratio (LER), and Earnings per Share (EPS). Table 2 provides detailed definitions for each variable.

Table 2. Independent variables (features) and relevant definitions.

Independent variables	Symbols	Definition
Prospectus characteristics		
Sector Code	SC	This is a label for the corresponding sector
Total Number of Offer Shares	TOS	This is the total number of shares that the company is offering for sale during the IPO. It can give an indication of the size of the IPO and the level of equity the company is willing to distribute to the public
Offer Price	OP	This is the price at which each share is offered during the IPO. It is set by the company (often in consultation with its investment bankers) based on a variety of factors including the company's valuation, market conditions, and the anticipated demand for the shares
Number of Substantial Shareholders	SH	This refers to the number of shareholders who hold a significant portion of the company's shares. What constitutes a "substantial" shareholding can vary, but it typically refers to shareholders who own a certain percentage (e.g., 5% or 10%) of the total issued shares
Subscription Quarter	SQ	This represents the fiscal quarter, which is one of the four three-month periods that make up a company's fiscal year. These quarters are denoted Q1, Q2, Q3, and Q4.
Individual Coverage	ind_sub%	This is an indication of how many times shares offered for sale to individual investors have been subscribed or applied for
Institutional Coverage	ins_cov%	This is an indication of how many times shares offered for sale to institutional investors have been subscribed or applied for
Financial ratios		
Net Profit Margin	NPM%	This ratio indicates the percentage of revenue that exceeds all of the company's costs, including both COGS and indirect expenses. It is calculated by subtracting all costs from revenue, then by dividing the result by revenue. A higher net profit margin indicates a more profitable company
Return on Assets (ROA)	ROA%	This ratio indicates how profitable a company is relative to its total assets. It is calculated by dividing net income by total assets
Current Assets to Current Liabilities (current_ratio)	CR	Also known as the Current Ratio, it measures a company's ability to pay off its short-term liabilities with its short-term assets. A higher ratio indicates better short-term financial health
Liabilities to Equity Ratio	LER	Also known as the Debt-to-Equity Ratio, this ratio compares a company's total liabilities to its shareholder equity. A high ratio suggests that the company has been aggressive in financing its growth with debt
Earnings per Share	EPS	This ratio is a portion of a company's profit allocated to each outstanding share of common stock. It's calculated by dividing net income by the number of outstanding shares. It serves as an indicator of a company's profitability

Preprocessing: In order to reduce computation time and data size, data preprocessing is essential. Preprocessing data for IPOs were "pre-processed and prepared for the model by cleaning, filtering, and transforming it to ensure its quality and consistency. The process consists of several steps, including identification and filling of missing values or removal of rows/columns with substantial missing data, and duplicates or typos Data were standardized and converted in a consistent format.

Data transformation: To prepare the dataset for analysis and ensure compliance with statistical assumptions, appropriate transformation methods were applied to each column based on its specific data properties (e.g., positive values, non-negative values, or mixed values). The transformations aimed to reduce skewness, stabilize variance, and enhance the suitability of variables for regression modeling. In this respect, *Box-Cox* transformation was applied to columns containing strictly positive values. This approach is well-suited for variables exhibiting right-skewed distributions as it effectively normalizes the data. $\log(1+x)$ transformation could be implemented on columns with non-negative values (including zero). The $\log(1+x)$ transformation is particularly effective for reducing skewness in distributions with a large range of non-negative values. *Square root* transformation is used to transform

columns containing a mix of positive and negative values. The square root transformation normalizes distributions while preserving the relative differences between positive and negative values.

Validation of transformations: To evaluate the effectiveness of the transformations, skewness values were calculated before and after applying the respective methods. A reduction in skewness indicated successful normalization of the variables. This step ensured the suitability of the transformed variables for robust statistical analysis.

Multicollinearity tests: Variance Inflation Factor (VIF) was employed to assess the degree of multicollinearity among the independent variables in the dataset. Multicollinearity can inflate standard errors of the coefficients, making it difficult to interpret the regression results. The VIF analysis ensures that the predictors are independent enough to yield stable and reliable estimates in the regression model.

Training and Testing: This step involved the division of the dataset into training and testing. For training, the split ratio was 90%, while for testing, it was 10%. When considering a limited dataset, the rationale for utilizing a 9:1 single split validation method can be supported by its computational efficiency, methodological simplicity, and the advantage of dedicating a greater portion of the data to model training. By avoiding the high variability and potential overfitting that are inherent in K-fold cross-validation due to multiple resampling, this method circumvents these issues. This is especially problematic considering the limited data available. Although potential biases and reduced generalizability are drawbacks of this validation strategy, it was considered suitable for the exploratory nature of the study and the requirement to optimize learning from a limited sample size while conserving computational resources.

Features engineering: Due to the presence of outliers and a large number of determinants, it has been particularly challenging to predict initial returns for IPOs. One potential solution for dealing with the large number of determinants in IPO return prediction is to use feature selection techniques. These techniques can help identify the most relevant and informative predictors, reducing the dimensionality of the problem and improving the model's performance. As part of the feature engineering process for multinomial logistic regression, eliminating correlated features involves the identification and removal of highly correlated features with the objective of improving model performance and interpretability.

Pipeline:

Notation: Training Dataset is represented by:

$\mathbf{S} = \{(\mathbf{x}^{(1)}, \mathbf{y}^{(1)}), (\mathbf{x}^{(2)}, \mathbf{y}^{(2)}), \dots, (\mathbf{x}^{(n)}, \mathbf{y}^{(n)})\}$, here $\mathbf{x}^{(i)} \in \mathbf{R}^d$ is a d -dimensional vector representing the i^{th} -example (or i^{th} -record of data) and $\mathbf{y}^{(i)} \in \mathbf{k}$ is the class label corresponding to the i^{th} -example. The number of examples is denoted by n .

The First Step of the procedure involves training the Multinomial Logistic Regression model. Model parameters denoted by $\theta \in \mathbf{R}^d$ are initialized with random values. Then, all the feature vectors in the dataset are pre-processed using the Normalization Transformation. Next, the hypothesis function $h_{\theta}(x)$ is evaluated, which involves taking the dot product of parameters θ and data x , and then applying Softmax non-linear activation function to the result.

Negative Log Likelihood is used as the Loss/Cost function $J(\theta)$. Gradient of the Loss function is taken, and Gradient Descent update rule is used to update the model parameters using the learning rate α . This process is repeated until the model converges and an optimal value for the model parameters is obtained.

In the second step, class probabilities are calculated to assign labels. Probability of each class $c \in k$ probability $P(y = c | x, \theta)$ is obtained using the trained model. The Probability values are compared with a predetermined Threshold. The class with value that exceeds the Threshold is assigned as the label.

Finally, the results are evaluated by computing a variety of metrics to analyze the performance of the model.

Algorithm 1: Multinomial Logistic Regression

Input:

Training Data $S = \{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \dots, (x^{(n)}, y^{(n)})\}$

Classes $k = \{1, 2, 3\}$

$x^{(i)} \in R^d, y^{(i)} \in k$

Number of Examples = n

Threshold = 0.5

Learning Rate $\alpha \in (0, 1)$, typically $\alpha = 0.01$

Procedure:
Step 1: Model Training.

Initialize Model Parameters $\theta \in R^d$ arbitrarily.

Normalize Data $x^{(i)} = \frac{x^{(i)}}{\sum_{j=1}^n |x^{(i)}|}$ for all $i \in \{1, 2, \dots, n\}$ ► transform () method

while J is not converged **do**

Hypothesis Function $h_{\theta_c}(x^{(i)}) = \frac{\exp(\theta_c^T x^{(i)})}{\sum_{j=1}^k \exp(\theta_j^T x^{(i)})}$ for all $c \in k$ ► fit () method

Negative Log Likelihood Loss $J(\theta) = \sum_{i=1}^n -\log h_{\theta_i}(x^{(i)})$

Update Parameters using Gradient Descent $\theta := \theta + \alpha \nabla J(\theta)$

return Trained Model $h_{\theta}(x)$

Step 2: Evaluating Class Probabilities and Assigning Labels.

for $c = 1, \dots, k$ **do**

Class Probability $P(y = c | x, \theta) = h_{\theta_c}(x^{(i)})$ ► predict_proba() method

if $P(y = c | x, \theta) > \text{Threshold}$ **then**

$y^{(i)} = c$

Output: Category $y^{(i)}$ corresponding to example $x^{(i)}$.

Application of Multinomial Logistic Regression (MLR): This is a method of classification known as Multinomial Logistics Regression (MLR) or Soft-Max Regression which generalizes multiclass logistic regression by allowing for more than two possible outcomes (Vryniotis, 2023). The purpose of this study was to use this algorithm to categorize data into three different categories which are “BELOW AVERAGE”, “AVERAGE”, and “ABOVE AVERAGE” to classify the performance of recently listed stocks with respect to their returns from their initial public offering (IPO) prices.

Assuming there are k classes, the formula for MLR is as follows:

$$P(y = c | x; \theta_1, \theta_2, \dots, \theta_k) = \frac{\exp(\theta_c^T x)}{\sum_{j=1}^k \exp(\theta_j^T x)} \quad (1)$$

where $\theta_1, \theta_2, \dots, \theta_k$ are the parameters.

In the above equation, it is shown that the probability of the third class is denoted as ‘c’, which is the probability of the third class. Using this formula, we could calculate the probability of the third class. In order to minimize the functions of the class, the values of θ_s are updated simultaneously. MLR was calculated using the following SoftMax Function, follows:

$$y|x; \theta_1, \theta_2, \dots, \theta_k \sim \text{Multinomial}(\phi_1, \phi_2, \dots, \phi_k) \quad (2)$$

where $\phi_j = h_{\theta_j}(x)$

There are 3 classes in this equation; therefore, $\phi_1, \phi_2, \dots, \phi_k$ are the lowest values for the features that belong to each class. In order to classify the comments on the basis of these values, the SoftMax classifies the comments into more than two classes.

The following equation (Equation 3) presents a formula for classifying a class into more than two groups according to MLR:

$$h_{\theta_c}(x) = \frac{\exp(\theta_c^T x)}{\sum_{j=1}^k \exp(\theta_j^T x)} \quad (3)$$

In this context, the logistic model with three categories has two logit functions:

(i) Logit Function for $Y = 0$ relative to logit function for $Y = 2$

(ii) Logit Function for $Y = 1$ relative to logit function for $Y = 2$

Category $Y = 2$ is called a *reference group*.

$$\log(p/1-p) = A + B_1X_1 + \dots + B_kX_k \quad (4)$$

$$\log(g(1)) = A_1 + B_{11}X_1 + \dots + B_{1k}X_k \quad (5)$$

$$\log(g(2)) = A_2 + B_{21}X_1 + \dots + B_{2k}X_k \quad (6)$$

$$\log(g(3)) = \log 1 = 0$$

Here are the results of Multilogistic regression using the formula.

$$f(1) = \frac{g(1)}{g(1)+g(2)+1} = \frac{e^{A_1+B_{11}X_1+\dots+B_{1k}X_k}}{e^{A_1+B_{11}X_1+\dots+B_{1k}X_k}+e^{A_2+B_{21}X_1+\dots+B_{2k}X_k}+1} \quad (7)$$

$$f(2) = \frac{g(2)}{g(1)+g(2)+1} = \frac{e^{A_2+B_{21}X_1+\dots+B_{2k}X_k}}{e^{A_1+B_{11}X_1+\dots+B_{1k}X_k}+e^{A_2+B_{21}X_1+\dots+B_{2k}X_k}+1} \quad (8)$$

$$f(3) = \frac{g(3)}{g(1)+g(2)+1} = \frac{1}{e^{A_1+B_{11}X_1+\dots+B_{1k}X_k}+e^{A_2+B_{21}X_1+\dots+B_{2k}X_k}+1} \quad (9)$$

We define $f(1)$ as the probability of BELOW AVERAGE, $f(2)$ as the probability of AVERAGE performance, and $f(3)$ as the probability of ABOVE AVERAGE. By putting the independent variable values into the equations above, we are able to obtain values ranging from 0 to 1, because $f(1) + f(2) + f(3) = 1$. It can be concluded that if the value of $f(1)$ exceeds 0.5, then it is classified as “BELOW AVERAGE”, or if $f(2)$ exceeds 0.5, then it is classified as “AVERAGE”, or if $f(3)$ exceeds 0.5, then it can be classified as “ABOVE AVERAGE”, and if all of these values are less than 0.5, then it is classified as “Unclassified”.

Performance metrics: In retrieving details from a database, precision and recall are also factors that are commonly used. The relationship between precision and recall is inversely proportional, which makes it necessary to be able to differentiate them through an effective classification method (Qutab

et al., 2022). Using Equations 10 and 11, we could determine the following measures for a classification task:

$$\text{Precision} = \frac{\text{True Positive (TP)}}{\text{True Positive (TP)} + \text{False Positive (FP)}} \quad (10)$$

$$\text{Recall} = \frac{\text{True Positive (TP)}}{\text{True Positive (TP)} + \text{False Negative (FN)}} \quad (11)$$

As part of the measurement of classification models, accuracy is a parameter that is taken into account. Essentially, accuracy is the number of predictions that were correctly predicted in the model based on the data and can be expressed, as follows in Equation 12:

$$\text{Accuracy} = \frac{\text{True Positive (TP)} + \text{True Negative (TN)}}{\text{True Positive (TP)} + \text{True Negative (TN)} + \text{False Negative (FN)} + \text{False Positive (FP)}} \quad (12)$$

F1-Score is a calculation that is identical to the calculation of the F-score or the F-measurement. It is usually regarded that precision and recall refer to the same number of positive results across the total, including not properly defined positive results. Recall is defined as the number of correctly recognized positives divided by the number of total samples classified as positives. The F1-score can be calculated as follows using Equation 13:

$$\text{F1-Score} = \frac{2 * (\text{Precision} * \text{Recall})}{(\text{Precision} + \text{Recall})} \quad (13)$$

ROC curves: To assess whether the multinomial logistic regression can discriminate multiple categories of outcome variables, receiver operating characteristic (ROC) curves were also used. A multinomial logistic regression ROC curve can be used to compare the overall predictive performance across all categories, as opposed to binary logistic regression, where the ROC curve plots true positives against false positives for one category. This is achieved by comparing each outcome category with the rest or by using cumulative ROC curves for models that contain three or more levels of outcome (DeCastro, 2019).

Confusion matrix: As used in computational classification, confusion matrices are also known as error matrices. To distinguish specific values in the test dataset, the output of a classification model (or classifier model) is often defined based on the values that it can distinguish. By doing so, it is possible to gain a better understanding of the performance of an algorithm. True positive (TP), True Negative (TN), False Negative (FN), and False positive (FP) values were used to measure the accuracy of the model.

Model fitting criteria: A pseudo R-squared can be used to measure a model's fit, but it does not convey as much information as a linear R-squared. A log-likelihood change from the null model essentially corresponds to the difference between the two models. A model with a higher pseudo-R-squared value will fit the data better than one with a lower pseudo-R-squared value ("Multinomial logistic regression | stata data analysis examples," 2023). In multinomial logistic regression, likelihood ratio tests are used to determine how significant the predictors in the researcher's model and their ability to explain the variance in the outcome variables are in the researcher's model ("Multinomial Logistic Regression," 2023).

Parameter estimates: An iterative maximum likelihood method is used to estimate regression coefficients (β s) using an iterate procedure.

Validation of the MLR model: To ensure that the model is capable of predicting, it is validated against independent data. A model fitting process usually begins by collecting an independent data set and validating it.

Ranking independent variables: The rank of the independent variables are plotted with respect to their magnitudes and significance using *matplotlib.pyplot library* in python after the model was fitted and the coefficients were retrieved.

Comparison with machine learning algorithms: Using Python, the MLR model is compared to other machine learning classification algorithms (Logistic Regression, Decision Tree Classifier, Random Forest Classifier, Support Vector Classifier (SVC), Gradient Boosting Classifier, K-Neighbours Classifier, Gaussian NB, XGB Classifier, Ada Boost Classifier) in terms of their accuracy levels after preparing data, fitting models, evaluating their performance, comparing them, and considering interpretability, complexity, and computational complexity.

The Integrated Development Environment (IDE): PyCharm is used as a tool to have all the work develop the model. The code was easily created, tested, debugged, refactored, and analyzed.

4. Experimental results

4.1. Data transformation and multicollinearity

Table 3 shows the comprehensive analysis of skewness, applied transformations, and VIF values of features. The skewness analysis revealed that several variables exhibited significant deviations from normality in their original forms. Specifically, variables such as TOS, OP, NPM%, CR, LER, EPS, and ind_sub% (explained in Table 2) displayed high positive skewness values (e.g., OP = 6.253, CR = 3.415, EPS = 4.107). After applying Box-Cox transformations, the skewness of these variables was significantly reduced, bringing them closer to normal distributions (e.g., TOS: 1.781 \rightarrow 0.065, OP: 6.253 \rightarrow -0.169, CR: 3.415 \rightarrow -0.024). In contrast, variables such as SC, SH, SQ, and ROA% showed relatively low skewness in their original forms (e.g., SC = 0.01863, SQ = 0.09496) and did not require transformations. This ensured the preservation of their original distributions without adversely impacting the model. Regarding the multicollinearity assessment, based on variance inflation factor (VIF) values, it indicated no severe multicollinearity among the independent variables. All VIF values were below the commonly accepted threshold of 5, with the highest values observed for EPS (1.859), OP (1.757), and SQ (1.598). These values confirm the independence of predictors, ensuring stable coefficient estimates in the multinomial logistic regression model. The low VIF values across all predictors suggest that multicollinearity is not a significant concern in the dataset. This finding is critical, as high multicollinearity can inflate standard errors and weaken the reliability of individual predictor coefficients. Variables such as OP and EPS, with relatively higher VIF values (1.757 and 1.859, respectively), should be carefully interpreted, although their VIF values remain well within acceptable limits. The combination of skewness correction and multicollinearity mitigation ensures the robustness of the multinomial logistic regression model. Based on the analysis, no independent variable needs to be removed from the dataset as both skewness corrections and VIF assessments indicate that all features are appropriately transformed and exhibit acceptable levels of multicollinearity, ensuring the robustness of the regression model.

Table 3. Comprehensive analysis of skewness, applied transformations, and VIF values of features.

Features	Skewness		Transformation	VIF
	Original skewness	Transformed skewness		
SC	0.01863	0.01863	No transformation	1.116
TOS	1.781388	0.065002	Box-Cox	1.276
OP	6.253283	-0.16866	Box-Cox	1.757
SH	0.463042	0.463042	No transformation	1.341
SQ	0.094963	0.094963	No transformation	1.598
NPM%	2.106773	-0.00446	Box-Cox	1.257
ROA%	0.986566	0.986566	No transformation	1.421
CR	3.415087	-0.02442	Box-Cox	1.123
LER	6.227062	-0.1286	Box-Cox	1.191
EPS	4.107133	-0.06596	Box-Cox	1.859
ind_sub%	3.551025	0.047192	Box-Cox	1.535
ins_cov%	1.673058	-0.023	Box-Cox	1.666

4.2. Model fitting

The fitting information for the model can be seen in Table 4. In order to determine whether a model has a better likelihood ratio than a null model, a likelihood ratio test was performed. The test is called the chi-square statistic, which is the difference between the $-2 \log$ -likelihoods of the two models, namely the Null and the Final model. The level of significance as less than 0.05, which shows a conclusion that the Final model outperforms the Null model. The results of the appropriateness test are also shown in Table 4, which contains model fitting information. Based on the likelihood ratio test, we could determine whether the model was better suited to fit the data than the null model. In the Null and Final models, the difference between $-2 \log$ -likelihoods is called the chi-square statistic. Considering the test significance level is less than 0.05, we could conclude that the Final model outperformed the Null model. This meant that the Final model was more appropriate to explain the data than the Null model. Finally, the results of the goodness-of-fit test indicated that the Final model is a good fit for the data. A $-2 \log$ -likelihood was calculated for each effect for a reduced model, which is a model without the effect. A significant value of 0.001 was calculated. The pseudo R^2 indicated the test results for variance of the model (Table 4). As a measure of how much of the variation in the response could be explained by a linear regression model, the R^2 statistic was used. Since it was not possible to calculate the R^2 statistic precisely for multinomial logistic regression models, these approximations were used. An increased pseudo R^2 statistic indicated that the model accounted for more variation to a maximum of one. In our test, the pseudo R^2 was calculated to be 0.395, which was remarkably greater than previously calculated one (Upadhyay et al., 2012). This revealed that our model outperformed other models with respect to the explanation of variation. This indicated that our multinomial logistic regression model is more reliable in predicting the target outcome than past models. This model can be used to make more accurate predictions, enabling better decision making.

Table 4. Model fitting information and pseudo R^2 square

Model	Model fitting criteria	Likelihood ratio tests			pseudo R^2 square
	$-2 \log$ Likelihood	Chi-square	df	Significance	McFadden
Intercept only	95.608	-	-	-	0.395
Final	57.840	37.769	14	0.001	

4.3. Likelihood ratio test results

Table 5 presents the results of likelihood ratio tests conducted for both non-transformed and transformed features. The findings indicate substantial differences in the contribution of individual predictors to the model before and after transformation, as discussed below. Test of significance of parameters was performed by likelihood ratio tests (Table 5). This test is used to compare two competing hypotheses, and is used to determine if the difference in parameters between two groups is statistically significant. A likelihood ratio test determines whether each effect contributes to the model. The test can also be used to identify which parameters have the greatest impact on the outcome of an experiment. The chi-square statistic is the difference between the final and reduced models in terms of -2 log likelihoods. An effect is removed from the final model to create a reduced model. All parameters of the effect are considered to be zero under the null hypothesis. We compute the -2 log likelihood for a reduced model, one that does not include the effect. A small significance value (e.g., less than 0.10) indicates that the effect contributes to the model.

Table 5. Likelihood ratio tests for non-transformed and transformed features.

Effect	Non-transformed features			Transformed features		
	-2 Log Likelihood	Chi-square	Sig.*	-2 Log Likelihood	Chi-square	Sig.*
Intercept	67.142	9.303	0.010	68.636	11.938	0.003
SC	64.540	6.701	0.035	60.636	3.963	0.138
TOS	58.420	0.581	0.748	60.580	3.883	0.144
OP	58.500	0.661	0.719	64.640	7.943	0.019
SH	59.000	1.161	0.560	57.540	0.843	0.656
SQ	70.620	12.781	0.002	67.900	11.203	0.004
NPM%	64.480	6.641	0.036	62.880	6.183	0.045
ROA%	63.960	6.121	0.047	59.400	2.703	0.259
CR	58.980	1.141	0.565	58.540	1.843	0.398
LER	62.940	5.101	0.078	61.820	5.123	0.077
EPS	63.500	5.661	0.059	56.920	0.223	0.895
ind_sub%	62.620	4.781	0.092	67.680	10.983	0.004
ins_cov%	58.400	0.561	0.755	63.420	6.723	0.035

* In the upcoming tables, the reduced model was developed by including only the significant features. This was done by excluding the insignificant features from the full model: TOS, OP, SH, CR, and ins_cov% for analyzing the non-transformed features, and SC, TOS, SH, ROA%, CR, and EPS for analyzing the transformed features.

According to our analysis, the features, TOS, OP, SH, CR, and ins_cov% showed non-significance ($p > 0.1$). Other variables, SC, TOS, SH, ROA%, CR and EPS remained non-significant post-transformation. The comparison between non-transformed and transformed features provided insights into the impact of data preprocessing on model performance. Notably, feature transformation generally enhanced the significance and interpretability of certain predictors, as observed for variables such as OP, ins_cov%, and the intercept. The improved likelihood values (-2 Log Likelihood) and chi-square statistics indicated better model fit with transformed features. However, certain variables (SC, ROA%, and EPS) lost their significance after transformation, suggesting that the transformations may reduce their relative contribution. This outcome could result from the underlying data distribution or the transformations applied, warranting further exploration. The sustained significance of SQ, NPM%, ER, and ind_sub across both models highlights their critical role in predicting outcomes. This consistency underscores their robustness and relevance regardless of preprocessing. The results

suggest that while transformations can improve model performance, they should be applied judiciously to avoid diminishing the impact of relevant predictors.

4.4. Model performance

Table 6. Performance metrics for full and reduced MLR models using non-transformed and transformed features.

Performance level	Non-transformed features				Transformed features			
	Precision	Recall	F1-Score	Support	Precision	Recall	F1-Score	Support
Full MLR model								
BELOW AVERAGE	0.750	0.500	0.600	12	0.700	0.583	0.636	12
AVERAGE	0.727	0.857	0.787	28	0.839	0.929	0.881	28
ABOVE AVERAGE	0.625	0.556	0.588	9	0.750	0.667	0.706	9
Accuracy				0.714				0.796
Macro avg	0.701	0.638	0.658	49	0.763	0.726	0.741	49
Weighted avg	0.714	0.714	0.705	49	0.788	0.796	0.789	49
Reduced MLR model								
BELOW AVERAGE	0.667	0.500	0.571	12	0.667	0.333	0.444	12
AVERAGE	0.706	0.857	0.774	28	0.676	0.893	0.769	28
ABOVE AVERAGE	0.667	0.444	0.533	9	0.667	0.444	0.533	9
Accuracy				0.694				0.673
Macro avg	0.678	0.600	0.626	49	0.670	0.557	0.582	49
Weighted avg	0.689	0.694	0.680	49	0.672	0.673	0.646	49

Multinomial Logistic Regression (MLR) was used to develop a classification model, using Python. A 90:10 split ratio was used to divide the dataset into training and testing. The rationale behind choosing a 90:10 split ratio is to allocate a larger portion of the dataset for training the model. On the one hand, by allocating 90% of the data for training, the model can learn patterns and relationships more effectively. The remaining 10% of the data was then used for validating the model's performance and evaluating its accuracy. On the other hand, a higher validation set percentage, for instance (70:30), provides more records for validating the model's performance, which can be beneficial for assessing its generalization ability but usually requires large datasets. Table 6 compares the performance of full and reduced multinomial logistic regression (MLR) models using both non-transformed and transformed features. In the full model, most of the parameters (7 out of 12) for non-transformed features and half of the parameters (6 out of 12) for transformed features were significant; however, the reduced model consisted only of significant coefficients for both non-transformed and transformed features (Table 5). To enable a thorough comparison, results were provided for both the full model—where some coefficients remained non-significant for both the original and transformed features—and the reduced model, which excluded the non-significant parameters. Performance metrics such as precision, recall, F1-score, and accuracy are presented for three performance categories: BELOW AVERAGE, AVERAGE, and ABOVE AVERAGE. Additionally, macro-averaged and weighted-averaged values summarize the overall model performance. The results of the analysis of the test data using MLR showed that the test data could be classified into three categories based on our model: Below average, average, and above average. Simply analyzing the accuracy of a model is insufficient for evaluating its performance. In addition, the precision, recall, and F1 values should be calculated to reveal how well the model can classify each category based on its predictions (Feng et al., 2022). In this respect, with the help of the *cross_validate* function from the *sklearn.model_selection* package

from Scikit-learn³⁴, we were able to obtain performance metrics such as accuracy, precision, recall, and F1 score for our model (Pedregosa et al., 2011).

Performance of the full MLR model can be seen in Table 6. The model achieved an overall accuracy of 0.714, with macro-averaged F1-score at 0.658 and weighted-averaged F1-score at 0.705. For the AVERAGE category, the model showed the highest recall (0.857) and F1-score (0.787) among all categories. This suggests that the model performs well in identifying IPOs with average performance. BELOW AVERAGE and ABOVE AVERAGE categories showed lower recall values (0.500 and 0.556, respectively), indicating weaker performance in identifying IPOs in these groups. After applying transformations, the accuracy improved to 0.796, with macro-averaged F1-score increasing to 0.741 and weighted-averaged F1-score improving to 0.789. Performance gains were observed across all categories, particularly in recall for ABOVE AVERAGE (0.667) and BELOW AVERAGE (0.583). The AVERAGE category maintained the highest recall (0.929) and F1-score (0.881), demonstrating consistent robustness in this category. These improvements highlight the positive impact of transformations in addressing skewness and stabilizing variance. These levels of accuracy seem to be higher than those reported in literature. Ni (2023) has reported accuracy levels ranging between 45 and 58% after training 6 machine learning models, namely Random Forests, Decision Tree, Naive Bayes, Logistic Regression, Light GBM, and a stack model to predict the direction of Hong Kong stocks IPO performances (positive, negative, or little change) in 30 days after their IPOs. Emidi et al. (2022) used classification trees, logistic regressions, and support vector machine algorithms to predict the outcome of IPOs, with a ground accuracy level of 0.69, 0.67, and 0.65, respectively. The comparable predictions of IPO returns performance were also reported in the literature (Baba et al., 2020; Huang et al., 2021; Kooli et al., 2007). The recall value calculated for the AVERAGE class was 0.857, which was significantly higher than the reported (0.77) value found by the logistic regression model used by Emidi et al. (2022) to predict IPO outcomes. To enhance the model's performance and minimize the risk of overfitting, the MLR model was simplified by excluding certain independent variables. Specifically, TOS, OP, SH, SQ, and CR were removed from the non-transformed features, while SC, TOS, SH, ROA%, CR, and EPS were excluded from the transformed features based on likelihood test results (Table 5). In this respect, the variables were carefully selected by avoiding dropping important variables that can lead to biased estimates for regression coefficients and inflated p-values (Ranganathan et al., 2017). Regarding the performance of the reduced MLR model, the reduced model achieved a slightly lower accuracy (0.694) compared to the full model. The macro-averaged F1-score was 0.626, and the weighted-averaged F1-score was 0.680. Similar to the full model, the AVERAGE category outperformed the other categories with recall (0.857) and F1-score (0.774), while the BELOW AVERAGE and ABOVE AVERAGE categories showed poorer recall values (0.500 and 0.444, respectively). With transformed features, the accuracy slightly declined to 0.673, and the macro-averaged F1-score decreased to 0.582. The weighted-averaged F1-score also drops to 0.646. Transformations had a mixed impact on the performance of the reduced model. Recall that the BELOW AVERAGE category worsened to 0.333, while the recall for the AVERAGE category remained high at 0.893. This suggests that the reduced model struggles to maintain predictive power across all categories when transformations are applied.

Applying feature transformations significantly improved the performance of the full MLR model, particularly in terms of accuracy and F1-scores. This demonstrates the importance of preprocessing steps like skewness correction to enhance model reliability and predictive power. However, the impact of transformations on the reduced model is less favorable, potentially due to the exclusion of key

predictors that may have been better suited for transformed data. The full model consistently outperformed the reduced model, both with and without transformations. This was particularly evident in metrics like macro-averaged F1-score (e.g., 0.741 vs. 0.582 for transformed features). The reduced model sacrificed some predictive accuracy for simplicity. The reduced model may be preferred in resource-constrained scenarios, but its limited performance in specific categories (e.g., BELOW AVERAGE recall of 0.333) suggests caution in its application. As far as category-specific observations were concerned, both models consistently performed best in the AVERAGE category, achieving high recall and F1-scores. This indicates that the models were well-suited to capturing mid-level IPO performance trends. The BELOW AVERAGE and ABOVE AVERAGE categories showed weaker recall across all configurations, suggesting room for improvement in distinguishing these less-represented categories.

4.5. Confusion matrix

The confusion matrices presented for both non-transformed and transformed features in Table 7 illustrate the classification performance of the full and reduced MLR models in predicting three categories: Below Average, Average, and Above Average. From the confusion matrix (3×3 matrix with nine cells), it can be seen that diagonal values are the TPs (correct predictions) predicted by the model, while other values corresponded to TNs, FNs, and FPs. The diagonal 3 cells were correctly classified, but the remaining 6 cells were misclassified.

Table 7. Confusion matrix of full and reduced MLR models presented as confusion matrices, comparing non-transformed and transformed features

Observed	Non-transformed features			Predicted				
	Predicted BELOW AVERAGE	AVERAGE	ABOVE AVERAGE	Percent Correct	Predicted BELOW AVERAGE	ABOVE AVERAGE	Percent Correct	
Full model								
BELOW AVERAGE	6	5	1	50	7	3	2	58.33
AVERAGE	2	24	2	85.71	2	26	0	92.86
ABOVE AVERAGE	0	4	5	55.56	1	2	6	66.67
Percent overall Predicted	16.33	67.35	16.33	71.43	20.41	63.27	16.33	79.60
Reduced model								
BELOW AVERAGE	6	6	0	50	4	7	1	33
AVERAGE	2	24	2	85.71	2	25	1	89.29
ABOVE AVERAGE	1	4	4	44.44	0	5	4	44.44
Percent overall Predicted	18.37	69.39	12.24	69.39	12.24	75.51	12.24	67.35

The diagonal values generally improved after transformation. The model achieved an overall prediction accuracy of 71.43%. The highest class-specific accuracy was for the “AVERAGE” category, with a correct classification rate of 85.71%. However, the model performed poorly in classifying the “BELOW AVERAGE” and “ABOVE AVERAGE” categories, with correct classification rates of 50% and 55.56%, respectively. After transformation, the overall prediction accuracy improved to 79.60%. The accuracy for the “AVERAGE” category slightly increased to 92.86%, highlighting the robustness of the transformation in correctly identifying this category. There was also an improvement in the “BELOW AVERAGE” category, where the correct classification rate increased from 50% to 58.33%. The performance for the “ABOVE AVERAGE” category improved significantly, reaching 66.67%.

The transformation resulted in a noticeable improvement in the overall accuracy (from 71.43% to 79.60%) and class-specific performance for the “BELOW AVERAGE” and “ABOVE AVERAGE” categories. This suggests that the transformation helped the model better capture relationships between features and the target variable, especially for categories with smaller sample sizes or higher misclassification rates.

Regarding the results for the reduced model, the reduced model without transformation achieved an overall accuracy of 69.39%. The “AVERAGE” category had the highest accuracy, with 85.71% correct classifications. However, the performance for the “BELOW AVERAGE” and “ABOVE AVERAGE” categories remained low, at 50% and 44.44%, respectively. After applying feature transformation, the overall accuracy decreased slightly to 67.35%. The accuracy for the “AVERAGE” category remained high, improving marginally to 89.29%. However, the classification accuracy for the “BELOW AVERAGE” category dropped from 50% to 33%, indicating that the transformation negatively impacted the model’s ability to identify this category. The performance for the “ABOVE AVERAGE” category also remained poor, without any change from 44%. In contrast to the full model, the transformation did not enhance the reduced model’s performance. Instead, the overall accuracy decreased, and the “BELOW AVERAGE” category suffered from reduced classification accuracy. This indicates that while the transformation improved the full model’s performance, it might have introduced complexity or noise that adversely affected the reduced model’s ability to generalize.

4.6. Results for parameter estimates

Table 8 presents the parameter estimates, including coefficients, standard errors, t-values, p-values, and confidence intervals, for both the full and reduced multiple linear regression (MLR) models using non-transformed and transformed features, allowing for a detailed comparison of variable significance and model performance across transformations and model simplifications. The coefficients for the transformed features tend to have different magnitudes compared to the non-transformed features. The Full Model, which includes all variables, experienced notable changes in coefficient magnitudes and statistical significance after transformations were applied. Variables such as Net Profit Margin (NPM%), Leverage Ratio (LER), Current Ratio (CR), and Earnings Per Share (EPS) were not significant before transformation ($P > |t| > 0.05$). Following the transformation, certain variables showed shifts in statistical significance, with some gaining or losing it, while their coefficients underwent minor adjustments; however, the statistical insignificance of a few variables remained unchanged. For instance; before transformation, the coefficient of EPS was -0.504 ($P > |t| = 0.062$), close to significance. After transformation, the coefficient dropped to -0.043 ($P > |t| = 0.967$), fully losing any statistical relevance. To summarize, after transformation, the full model exhibited a general decrease in coefficient magnitudes for significant variables with increased standard errors, leading to reduced statistical power and loss of significance for variables such as SC and SQ, which were previously significant.

Table 8. Parameter estimates for the full and reduced MLR models using non-transformed and transformed features

Performance level*	Non-transformed features					Transformed features						
	Coef.	Std. Err.	t	P> t	[0.025	0.975]	Coef.	Std. Err.	t	P> t	[0.025	0.975]
FULL MODEL												
CLASS = 0 ^a												
constant	5.013	3.961	1.266	0.206	-2.749	12.775	-8.511	8.196	-1.038	0.299	-24.575	7.553
SC	0.555	0.262	2.123	0.034	0.043	1.068	0.353	0.200	1.763	0.078	-0.039	0.746
TOS	-0.003	0.025	-0.108	0.914	-0.052	0.047	0.597	0.595	1.004	0.315	-0.569	1.763
OP	-0.005	0.025	-0.204	0.838	-0.053	0.043	1.764	2.051	0.860	0.390	-2.257	5.784
SH	-0.348	0.334	-1.042	0.297	-1.002	0.306	0.052	0.291	0.178	0.859	-0.519	0.622
SQ	-1.896	0.955	-1.986	0.047	-3.767	-0.024	-0.800	0.658	-1.216	0.224	-2.089	0.489
NPM%	-0.114	0.071	-1.614	0.107	-0.253	0.025	-0.486	0.929	-0.523	0.601	-2.308	1.336
ROA%	0.216	0.126	1.714	0.087	-0.031	0.463	0.094	0.090	1.045	0.296	-0.083	0.272
CR	0.289	0.341	0.848	0.396	-0.379	0.958	-0.055	0.868	-0.063	0.950	-1.757	1.647
LER	-0.149	0.329	-0.452	0.651	-0.794	0.496	-0.574	0.742	-0.774	0.439	-2.028	0.880
EPS	-0.504	0.270	-1.870	0.062	-1.033	0.024	-0.043	1.051	-0.041	0.967	-2.102	2.016
ind_sub%	0.211	0.199	1.061	0.289	-0.179	0.602	2.827	1.329	2.127	0.033	0.222	5.432
ins_cov%	-0.009	0.020	-0.456	0.648	-0.049	0.031	-0.500	0.359	-1.395	0.163	-1.203	0.202
CLASS = 1 ^b												
constant	17.241	7.101	2.428	0.015	3.324	31.158	42.986	24.946	1.723	0.085	-5.907	91.879
SC	0.446	0.286	1.558	0.119	-0.115	1.007	0.238	0.265	0.897	0.370	-0.281	0.756
TOS	0.019	0.032	0.598	0.550	-0.044	0.082	-1.226	1.500	-0.818	0.414	-4.166	1.714
OP	-0.028	0.038	-0.747	0.455	-0.103	0.046	-6.850	4.072	-1.682	0.093	-14.831	1.130
SH	-0.265	0.340	-0.778	0.437	-0.932	0.402	-0.224	0.360	-0.624	0.533	-0.930	0.481
SQ	-3.567	1.349	-2.644	0.008	-6.211	-0.923	-3.523	1.592	-2.214	0.027	-6.642	-0.404
NPM%	-0.256	0.118	-2.167	0.030	-0.488	-0.025	-3.667	1.798	-2.040	0.041	-7.191	-0.144
ROA%	-0.004	0.186	-0.023	0.982	-0.370	0.361	-0.155	0.224	-0.693	0.488	-0.595	0.284
CR	0.279	0.418	0.667	0.505	-0.541	1.098	2.156	1.881	1.146	0.252	-1.530	5.842
LER	-2.827	1.433	-1.973	0.049	-5.636	-0.019	-4.359	2.462	-1.770	0.077	-9.185	0.467
EPS	-0.569	0.331	-1.719	0.086	-1.217	0.080	-1.275	2.849	-0.448	0.654	-6.859	4.309
ind_sub%	0.076	0.226	0.337	0.736	-0.367	0.520	-1.391	2.635	-0.528	0.598	-6.556	3.773
ins_cov%	0.007	0.032	0.220	0.826	-0.055	0.069	0.759	0.693	1.095	0.274	-0.600	2.118
REDUCED MODEL												
CLASS = 0 ^a												
Constant	2.429	2.307	1.053	0.292	-2.092	6.950	-3.384	5.206	-0.650	0.516	-13.588	6.821
SC	0.421	0.195	2.155	0.031	0.038	0.804	-	-	-	-	-	-
OP	-	-	-	-	-	-	1.089	1.316	0.827	0.408	-1.490	3.668
SQ	-1.351	0.650	-2.079	0.038	-2.625	-0.077	-0.614	0.451	-1.363	0.173	-1.498	0.269
NPM%	-0.092	0.049	-1.878	0.060	-0.189	0.004	0.322	0.698	0.461	0.645	-1.046	1.689
ROA%	0.187	0.096	1.953	0.051	-0.001	0.375	-	-	-	-	-	-
LER	-0.061	0.270	-0.227	0.820	-0.590	0.468	0.322	0.698	0.461	0.645	-1.046	1.689
EPS	-0.345	0.186	-1.855	0.064	-0.710	0.019	-	-	-	-	-	-
ind_sub%	0.130	0.083	1.560	0.119	-0.033	0.293	1.430	0.863	1.657	0.097	-0.261	3.122
ins_cov%	-	-	-	-	-	-	-0.165	0.238	-0.694	0.488	-0.632	0.302
CLASS = 1 ^b												
constant	12.054	4.679	2.576	0.010	2.884	21.225	21.110	10.404	2.029	0.042	0.719	41.501
SC	0.272	0.233	1.168	0.243	-0.185	0.730	-	-	-	-	-	-
OP	-	-	-	-	-	-	-3.740	2.190	-1.708	0.088	-8.033	0.553
SQ	-2.495	0.868	-2.873	0.004	-4.197	-0.793	-2.218	0.987	-2.246	0.025	-4.153	-0.283
NPM%	-0.158	0.073	-2.158	0.031	-0.302	-0.014	-2.308	1.271	-1.816	0.069	-4.798	0.183
ROA%	-0.033	0.162	-0.204	0.839	-0.351	0.285	-	-	-	-	-	-
LER	-2.360	1.152	-2.048	0.041	-4.618	-0.101	-1.856	0.955	-1.944	0.052	-3.727	0.015
EPS	-0.465	0.291	-1.599	0.110	-1.035	0.105	-	-	-	-	-	-
ind_sub%	0.037	0.110	0.341	0.733	-0.178	0.253	-1.030	1.602	-0.643	0.520	-4.169	2.110
ins_cov%	-	-	-	-	-	-	0.646	0.510	1.266	0.205	-0.354	1.645

* The reference category is “ABOVE AVERAGE”.

^{a,b} “BELOW AVERAGE” and “AVERAGE”, respectively.

The Reduced Model, which included a subset of variables, showed different patterns compared to the Full Model. Some variables retained significance after transformation, while others experienced

changes in magnitude or direction. SQ and NPM retained their significance after transformation, with SC showing a stronger effect. LER reversed its direction but remained significant. Some variables (e.g., NPM%) lost significance after transformation. Magnitudes and standard errors increased for most variables, potentially affecting confidence intervals.

Transformation had a noticeable effect on the coefficients, standard errors, and significance levels. In full model, transformation caused most variables to lose significance, likely due to increased standard errors and reduced coefficients. In reduced model, significant variables (e.g., SQ, NPM, and LER) generally retained their significance, with some becoming more impactful (e.g., SC). The reduced model demonstrated better retention of significance for key variables, with some coefficients becoming stronger after transformation. These findings highlight the importance of carefully evaluating the impact of data transformations on regression models as they can influence both the interpretability and reliability of the results.

The process of estimating logistic regression equations involves finding the values of the coefficients that maximize the likelihood of observing the given data. This is done by iteratively adjusting the coefficients until the model's predictions align as closely as possible with the actual outcomes. The maximum likelihood estimation allows us to determine the relationship between the independent variables and the probability of belonging to a particular class. In order to classify companies, the following multi logistic regression equations were estimated using the maximum likelihood estimation and are provided as the examples shown below:

for “BELOW AVERAGE” performance with respect to “ABOVE” performance; $Y_{BA/U}$:

$$Y_{BA/U} = 5.013 + 0.555 SC - 0.003 TOS - 0.005 OP - 0.348 SH - 1.896 SQ - 0.114 NPM\% + 0.216 ROA\% + 0.289 CR - 0.149 LER - 0.504 EPS + 0.211 ind_sub\% - 0.009 ins_cov\% \quad (14)$$

for “AVERAGE” performance with respect to “ABOVE” performance; $Y_{A/U}$:

$$Y_{A/U} = 17.241 + 0.446 SC + 0.019 TOS - 0.028 OP - 0.265 SH - 3.567 SQ - 0.256 NPM\% - 0.004 ROA\% + 0.279 CR - 2.827 LER - 0.569 EPS + 0.076 ind_sub\% - 0.007 ins_cov\% \quad (15)$$

4.7. Ranking independent variables

Variable importance in a model can greatly impact the predictions and insights derived from it. One strategy for interpreting variable importance rankings is to focus on the top-ranked variables and consider them as the most influential factors in the model's predictions. Another strategy is to compare the rankings across different models or datasets to identify consistent patterns of importance. Utilizing variable importance rankings can help prioritize resources and efforts towards the most impactful variables for further analysis or intervention. Figure 1 shows ranking independent variables in full model with respect to their corresponding levels of importance on the IPO performance level for two scenarios: (A) Untransformed features and (B) transformed features. In scenario (A), the feature 'SQ' demonstrated the highest chi-square value, indicating its significant contribution to the model's prediction. Other important predictors included 'Intercept', 'SC', and financial performance metrics such as 'NPM%' and 'ROA%'. However, features like 'TOS', 'OP', and 'ins_cov%' exhibited relatively lower chi-square values, suggesting a minor impact. In scenario (B), where the features were transformed, a notable shift in feature importance was observed. 'Intercept', 'SQ', and 'ind_sub%' emerged as the most influential features, whereas 'EPS' and 'SH' displayed minimal chi-square values,

indicating their reduced relevance in the transformed feature space. Interestingly, `ins_cov%`, which was among the least significant features in scenario (A), gained higher importance after transformation, reflecting the potential influence of feature engineering. Across both scenarios, a consistent trend was observed for a few features (`SQ` and `Intercept`), indicating their robustness and relevance regardless of feature transformation.

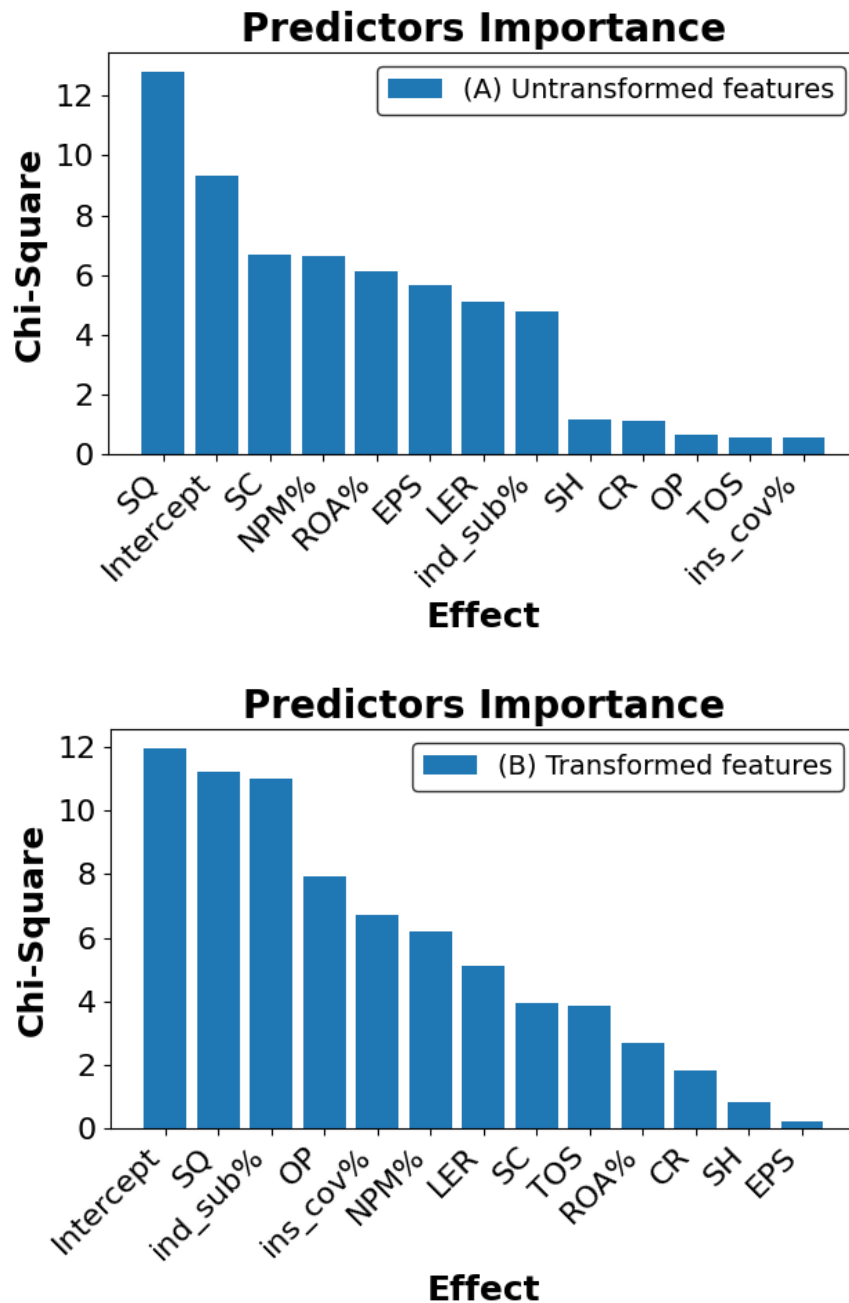


Figure 1. Comparison of feature importance based on chi-square values for (A) untransformed features and (B) transformed features.

The comparative analysis of feature importance highlights the impact of feature transformation on the predictive model's interpretability and performance. The higher chi-square values for `SQ` and

'Intercept' across both scenarios indicate their critical role in explaining the dependent variable, demonstrating stability in their predictive contributions. This stability suggests that these variables carry inherent predictive power and are less sensitive to transformation techniques. The increase in importance for 'ins_cov%' after transformation suggests that feature engineering can enhance the relevance of certain predictors, possibly by reducing noise or emphasizing non-linear relationships. This finding underlines the value of preprocessing steps in improving model accuracy and feature interpretability. Conversely, the reduced importance of features such as 'EPS' in the transformed scenario points to the potential loss of information during the transformation process. This emphasizes the need for a balanced approach to feature transformation, ensuring that critical predictive information is retained while improving overall model performance. Overall, this analysis highlights the importance of both raw and transformed features in constructing robust predictive models. While feature engineering can enhance certain predictors' importance, careful evaluation is required to preserve critical predictive relationships in the data.

Our multinomial logistic regression analysis reveals that the Subscription Quarter, Sector Code, and Net Profit Margin best predicted IPO performance. Subscription Quarter's prominence may reflect cyclical market dynamics and investor behaviour, where seasonal effects or economic indicators cause different investment patterns in certain fiscal periods. Sector Code may influence investor demand due to industry trends, investor risk profiles, and historical sector performance. Finally, Net Profit Margin emphasizes the importance of profitability as a key metric of a company's operational efficiency and growth potential, which investors consider when valuing new market entrants. These findings suggest that temporal market characteristics, sector dynamics, and fundamental financial health metrics drive IPO returns. However, market mechanisms and investor dynamics explain the marginal effect of Offer Price, Total Number of Offer Shares, and Individual Coverage on IPO returns. The Offer Price often matches market data, which may reduce its predictive power. IPO returns may not be affected by the Total Number of Offer Shares because the market can absorb different amounts of new shares without price changes. Individual coverage indicates retail investor interest, but institutional investors typically influence a stock's trajectory more. These factors suggest that market sentiment, sectoral shifts, and intrinsic financial health post-IPO may have a greater impact on short-term performance after listing.

Prediction of the success or performance of initial public offerings (IPOs) in stock exchanges is a complex task that involves analyzing various factors. However, significance order of these features can vary depending on the market conditions, economic climate, and industry trends. Therefore, a holistic approach considering multiple factors is recommended for a comprehensive IPO analysis. Additionally, market sentiment and macroeconomic factors can play a significant role in IPO performance. However, based on common financial analysis principles and our results obtained, it is possible to offer a general overview of each feature's potential significance.

4.8. Validation results

Table 9 provides validation results for the full and reduced MLR (Multinomial Logistic Regression) models using non-transformed and transformed features. The analysis evaluates the classification performance for six companies under different feature transformation scenarios. For non-transformed features, the full model achieved a perfect classification accuracy of 100%, correctly classifying all six companies. The reduced model demonstrated slightly lower performance, achieving 83.33% accuracy, with one misclassification (Company 3 was unclassified). The 83.33% accuracy of

the reduced model indicates that it is able to predict the outcome correctly for the majority of the test samples. Considering that logistic regression models do not require hyperparameter tuning and are capable of handling even a limited number of variables (Emidi et al., 2022), this accuracy level should be viewed as good. For transformed features, the full model exhibited a classification accuracy of 66.67%, with two companies (Company 1 and Company 5) misclassified as different performance levels. The reduced model, on the other hand, showed further performance decline, achieving an accuracy of 50.00%, with three companies (Companies 1, 3, and 5) unclassified or misclassified. Notably, probabilities of performance levels ($f(0)$, $f(1)$, and $f(2)$) varied significantly between models and feature transformations, reflecting the impact of feature engineering on the underlying data structure and model predictions.

The comparison between the full and reduced models highlights the trade-offs between model complexity and predictive accuracy. Using non-transformed features, the full model demonstrated superior performance, achieving perfect classification accuracy. This suggests that the inclusion of all predictors, without feature transformation, captured the underlying patterns effectively, ensuring precise classification. In contrast, slightly lower accuracy (83.33%) of the reduced model indicates that reducing the feature set can lead to a minor loss of predictive power. However, this reduction may still be acceptable in scenarios where simplicity and computational efficiency are prioritized over perfect accuracy. The transformed features yielded lower classification accuracies for both full and reduced models, suggesting that feature transformation introduced complexities that diminished the ability of models to distinguish performance levels. While transformation techniques are often employed to improve model generalizability and reduce overfitting, the results in this case indicate potential information loss or distortion caused by the transformation process. The performance of reduced model with transformed features (50.00%) underscores the importance of careful feature selection and transformation. Simplified models, while computationally efficient, may require additional preprocessing or feature engineering to retain predictive accuracy, especially when working with transformed data.

Table 6 shows that when transformed features were used, the full MLR model achieved higher precision, recall, and F1 scores, especially for the “Average” and “Above Average” performance levels. The overall accuracy improved from 71.43% (non-transformed) to 79.60% (transformed) for the full model. From confusion matrices in Table 7, it is clear that, for transformed features, the full model predicted a greater number of instances correctly across the three performance levels, particularly for the “Average” category, where accuracy reached 92.86%. However, in Table 9, validation accuracies for the full and reduced models decreased significantly after feature transformation. The validation accuracy of the full model dropped from 100% to 66.67%, and accuracy of the reduced model dropped further to 50.00% after transformation. At first glance, the results presented in Table 9 may appear to differ from those shown in Tables 6 and 7. The differences arises from differences in evaluation contexts and the influence of transformations on the specific data partitions used for validation. These variations can likely be explained by the following factors:

- ✓ Tables 6 and 7 focus on performance metrics such as precision, recall, and F1 scores, which provide detailed insights into classification performance for each class. These metrics evaluate how well the model identifies specific categories based on true positives, false positives, and false negatives.
- ✓ Table 9, on the other hand, reports validation accuracies (the proportion of correctly classified instances during validation), which depend heavily on how the validation data aligns with

learned patterns of the model.

- ✓ Feature transformation often introduces non-linear relationships or alters the data distribution, which can improve generalization for unseen data but may reduce the model's ability to perfectly classify instances in the validation set.
- ✓ In Tables 6 and 7, the transformed features likely enhanced the separation between performance levels in the training set, leading to better precision, recall, and F1 scores.
- ✓ In Table 9, the drop in validation accuracy suggests that the transformed features may have overfit specific patterns in the training set or failed to generalize well to the validation set.
- ✓ The reduced model showed greater sensitivity to feature transformations, as evidenced by the significant drop in both accuracy (Table 9) and F1 scores (Table 6). This could be due to the reduced number of predictors being less capable of capturing complex relationships introduced by transformations.
- ✓ The validation dataset may include instances with feature distributions that differ from the transformed training data, reducing ability of the model to classify correctly during validation (Table 9). This mismatch does not significantly affect the results in Tables 6 and 7, where performance was evaluated on the training set.

The contrasting outcomes highlight the trade-offs associated with feature transformations. Tables 6 and 7 emphasize within-sample performance, where transformations improved class separability and boosted precision, recall, and F1 scores. Table 9 reflects out-of-sample validation performance, where transformations reduced accuracy due to potential overfitting or misalignment with feature distributions of the validation set. This underscores the importance of carefully validating feature transformations and ensuring they generalize across all data partitions. These findings highlight the need for a balanced approach to model development, emphasizing the careful consideration of feature transformations and model complexity to optimize performance. The superior performance of non-transformed features in this study underscores their suitability for the given dataset and context. Overall, our study proves that, by having fewer variables, logistic regression models can effectively estimate the relationships between the variables and the outcome, leading to more accurate predictions without the need for extensive hyperparameter tuning.

Table 9. Validation results for the full and reduced MLR models using non-transformed and transformed features.

Full model	Non-transformed features			Validation results			Transformed features			Validation results		
	Probabilities of performance levels						Probabilities of performance levels					
	f(0)	f(1)	f(2)	Observed class	Predicted class	Result	f(0)	f(1)	f(2)	Observed class	Predicted class	Result
Company 1	0.9063	0.0933	0.0004	0	0	Classified	0.279	0.146	0.575	0	2	Unclassified
Company 2	0	0.8742	0.1257	1	1	Classified	0.0021	0.6962	0.3018	1	1	Classified
Company 3	0.4067	0.5933	0	1	1	Classified	0.0203	0.9797	0	1	1	Classified
Company 4	0.0319	0.0847	0.8834	2	2	Classified	0.0507	0.0512	0.8981	2	2	Classified
Company 5	0.9995	0.0005	0	0	0	Classified	0.4891	0.5101	0.0009	0	1	Unclassified
Company 6	1	0	0	0	0	Classified	0.7083	0.2917	0	0	0	Classified
						100 %						66.67 %
Reduced model												
Company 1	0.9572	0.0407	0.0021	0	0	Classified	0.3988	0.5091	0.0921	0	1	Unclassified
Company 2	0.0005	0.9099	0.0897	1	1	Classified	0.017	0.7516	0.2314	1	1	Classified
Company 3	0.649	0.35	0.0009	1	0	Unclassified	0.0613	0.9387	0.0001	1	1	Classified
Company 4	0.0567	0.1559	0.7874	2	2	Classified	0.1018	0.1885	0.7097	2	2	Classified
Company 5	0.9968	0.0032	0	0	0	Classified	0.4701	0.5262	0.0037	0	1	Unclassified
Company 6	0.9997	0.0003	0	0	0	Classified	0.2611	0.7389	0	0	1	Unclassified
						83.33 %						50.00 %

Table 10. Comparison of performance metrics of MLR model with those of machine learning classification algorithms.

Models	Performance metrics									
	Precision				Recall			F1 score		
	Accura	BELOW	AVERA	ABOVE	BELOW	AVERA	ABOVE	BELOW	AVERA	ABOVE
	cy	AVERAGE	GE	AVERAGE	AVERAGE	GE	AVERAGE	AVERAGE	GE	AVERAGE
MLR model	0.714	0.750	0.727	0.625	0.500	0.857	0.556	0.600	0.787	0.588
Logistic Regression	0.545	1	0.556	0	0.333	0.833	0	0.5	0.667	0
Decision Tree Classifier	0.363	0.4	0.5	1	0.667	0.5	0	0.5	0.5	0
Random Forest Classifier	0.545	0	0.375	0	0	0.5	0	0	0.429	0
Support Vector Classifier (SVC)	0.545	1	0.545	1	0	1	0	0	0.706	0
Gradient Boosting Classifier	0.272	0	0.5	1	0	0.5	1	0	0.5	1
Gaussian NB	0.272	0.5	0.667	0.333	0.333	0.667	0.5	0.4	0.667	0.4
K-Neighbor's Classifier	0.545	1	0.6	1	0	1	0.5	0	0.75	0.667
XGB Classifier	0.272	0	0.375	1	0	0.5	0	0	0.429	0
Ada Boost Classifier	0.545	0	0.5	1	0	0.833	0	0	0.625	0

4.9. Comparison of MLR accuracy with those of machine learning algorithms

Table 10 compares the performance metrics of the Multinomial Logistic Regression (MLR) model with various machine learning classification algorithms in terms of accuracy, precision, recall, and F1-score across three performance levels: “Below Average,” “Average,” and “Above Average.” The MLR model achieved the highest overall accuracy at 71.4%, outperforming all other models. Logistic Regression, Random Forest, Support Vector Classifier (SVC), K-Neighbors Classifier, and Ada Boost classifiers shared a lower accuracy of 54.5%. Gradient Boosting, Gaussian Naive Bayes (NB), Decision Tree, and XGB classifiers demonstrated even lower accuracy levels, ranging from 27.2% to 36.3%. Regarding precision, for the “Below Average” category, the MLR model achieved a precision of 0.750, which was surpassed only by Logistic Regression (1.000) and K-Neighbors Classifier (1.000). For the “Average” category, the MLR model achieved 0.727, outperforming all other models except the K-Neighbors Classifier (0.600) and Gaussian NB (0.667). For the “Above Average” category, the MLR model achieved a moderate precision of 0.625, which was comparable to Gradient Boosting (1.000) and Ada Boost (1.000), but higher than the other models. As for recall values, The MLR model showed a very good recall for the “Average” category (0.857), but moderate recall for the “Below Average” (0.500) and “Above Average” (0.556) categories. Logistic Regression and SVC outperformed for the “Average” category (0.833 and 1.000, respectively), but failed to recall any cases for the “Above Average” category. Other models, including Random Forest and SVC, failed to effectively recall instances for several categories. As far as F1-scores were concerned, the MLR model delivered a balanced F1-score across categories, achieving 0.600 (“Below Average”), 0.787 (“Average”), and 0.588 (“Above Average”). In contrast, most other models struggled to deliver balanced F1-scores, with Logistic Regression and K-Neighbors performing moderately well for some categories but failing for others.

The results from Table 10 highlight the superiority of the Multinomial Logistic Regression (MLR) model over other machine learning classifiers for the dataset under consideration, particularly when overall accuracy and balanced performance across categories are prioritized. The MLR model demonstrated the highest overall accuracy (71.4%), suggesting that it effectively captures the relationships in the dataset and generalizes well across all performance levels. The model also achieved balanced performance across all metrics (precision, recall, and F1-score), making it a reliable choice for datasets where consistent classification across categories is essential. Logistic Regression performed well for the “Below Average” and “Average” categories but failed entirely for the “Above Average” category, as indicated by its F1-score of 0.000 in that category. Decision Tree, Gradient Boosting, XGB, and Gaussian NB classifiers struggled with overall accuracy and failed to maintain consistent performance across categories, likely due to the nature of the dataset and insufficient generalization. K-Neighbors Classifier and Ada Boost showed some promise with high precision in certain categories, but their recall and F1-scores indicate poor performance in other areas, leading to overall subpar accuracy. The results suggest that the MLR model is particularly well-suited for datasets with multiple performance levels and complex categorical relationships. Its ability to balance precision, recall, and F1-score across categories makes it a robust choice. While some machine learning models (e.g., Logistic Regression, K-Neighbors Classifier) show potential in specific categories, their lack of generalization across all performance levels makes them less reliable for comprehensive classification tasks. Given the results, the MLR model should be the preferred choice for classification tasks on this dataset. It not only delivers the highest overall accuracy but also maintains balanced performance across all categories, making it a robust and interpretable option compared to more complex machine learning algorithms.

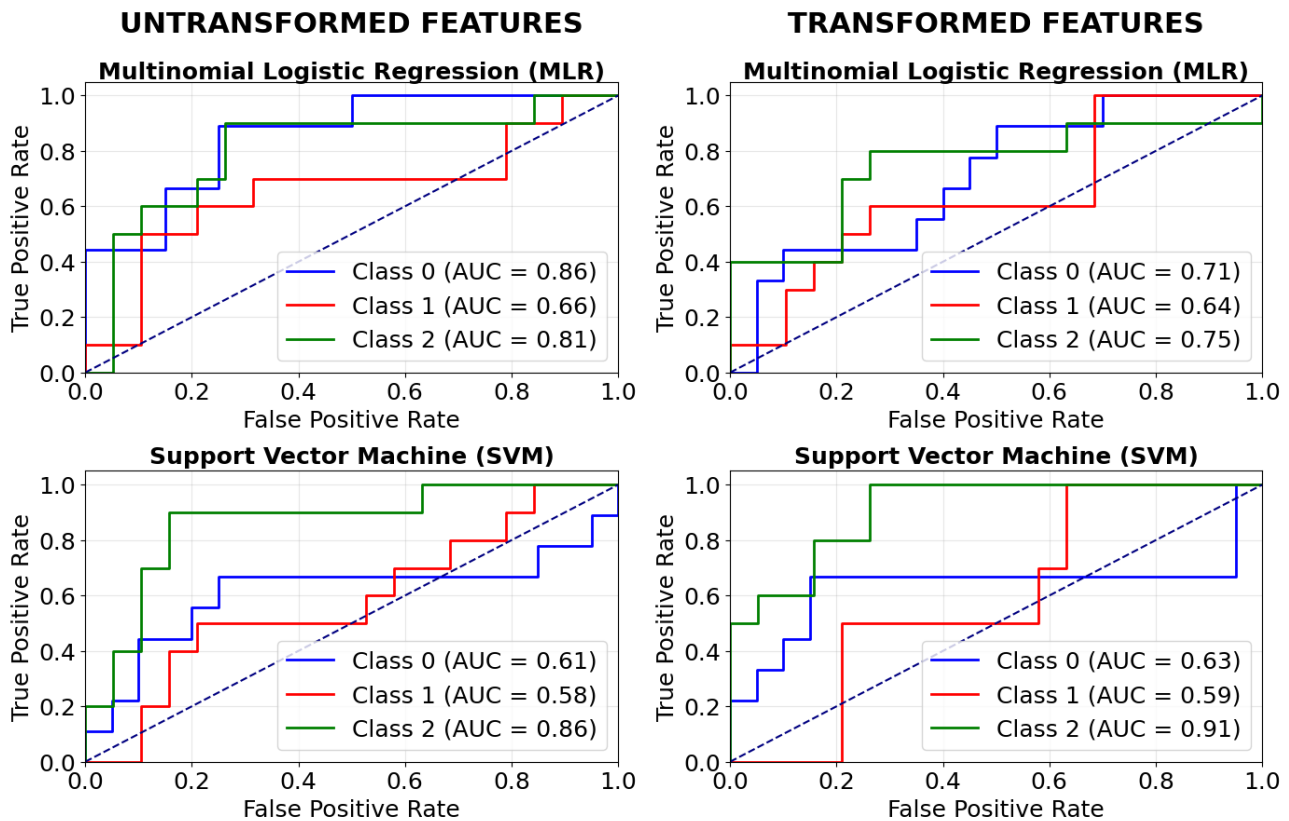


Figure 2. Comparison of Receiver Operating Characteristic (ROC) curves for Multinomial Logistic Regression (MLR) and Support Vector Machine (SVM) models under two feature scenarios: untransformed features (left column) and transformed features (right column). The plots demonstrate the True Positive Rate (TPR) versus False Positive Rate (FPR) for each class, with the corresponding Area Under the Curve (AUC) values indicated in the legend. Classes 0, 1 and 2 represent “BELOW AVERAGE”, “AVERAGE”, and “ABOVE AVERAGE”, respectively.

While the Multinomial Logistic Regression (MLR) model remains the best performer, SVC demonstrates superiority over the remaining machine learning models by achieving better accuracy, balanced precision, and recall for most categories. Its ability to generalize for the “Below Average” and “Average” categories highlights its robustness and suitability for this dataset, especially when compared to tree-based models and other classifiers; therefore, SVC was selected for comparison with MLR based on ROC curve accuracy values. Figure 2 presents Receiver Operating Characteristic (ROC) curves for the Multinomial Logistic Regression (MLR) and Support Vector Machine (SVM) models under two feature scenarios: untransformed features (left block) and transformed features (right block). The Area Under the Curve (AUC) values were calculated for three classes: Class 0 (“Below Average”), Class 1 (“Average”), and Class 2 (“Above Average”). The MLR model demonstrated very good performance with untransformed features. The AUC level falls between 0.65 and 0.9, it should be considered acceptable (Hosmer Jr et al., 2013). Comparable AUC values were reported for logistic regression used to predict IPO outcomes (Emidi et al., 2022). There are several factors that may have contributed to MLR’s superior performance. One reason is that it is designed to handle categorical outcomes, making it suitable for predicting the performance categories of IPO returns. Another possible reason is that MLR is better suited for handling linear relationships between the variables, which could be the case in this particular dataset. Additionally, MLR may have been able to capture

more complex interactions and dependencies among the features, allowing it to make more accurate predictions. However, this advantage was lost after feature transformation, as the AUC for Class 0 dropped significantly to 0.71, while the performance for Class 1 remained largely unchanged. Specifically, for Class 0, the AUC dropped from 0.86 (untransformed features) to 0.71 (transformed features), indicating a reduction in the model's ability to discriminate between Class 0 and the other classes after transformation. Similarly, for Class 2, the AUC decreased from 0.81 to 0.75, suggesting that the transformation diminished the separability of this class. Conversely, the AUC for Class 1 remained relatively stable, showing a minor change from 0.66 (untransformed) to 0.64 (transformed). These results suggest that while feature transformation can standardize the input data, it may also obscure certain feature relationships critical for effective classification by MLR. In contrast to MLR, the feature transformation had a generally positive effect on the performance of SVM. Notably, for Class 2, the AUC improved from 0.86 (untransformed features) to 0.91 (transformed features), indicating enhanced class separability and improved classification accuracy for this class. For Class 0, a slight improvement was observed, with the AUC increasing from 0.61 to 0.63. Similarly, for Class 1, the AUC showed a marginal increase from 0.58 to 0.59. These improvements suggest that SVM benefitted from the feature transformation, likely due to its ability to handle non-linear relationships more effectively when features are scaled or normalized.

A comparative analysis of the models before and after feature transformation revealed distinct trends. Using untransformed features, MLR outperformed SVM for Class 0 (AUC of 0.86 vs. 0.61) and Class 1 (AUC of 0.66 vs. 0.58), while both models performed similarly for Class 2 (AUC of 0.81 for MLR vs. 0.86 for SVM). After feature transformation, SVM outperformed MLR for Class 2, achieving an AUC of 0.91 higher than 0.75 for MLR. However, for Class 0 and Class 1, MLR demonstrated higher AUC values (0.71 and 0.64, respectively) than SVM (0.63 and 0.59, respectively). These results indicate that while SVM benefitted from feature transformation, particularly for Class 2, MLR retained its advantage for the other classes. Across both models and feature scenarios, Class 2 consistently exhibited the highest classification performance, with AUC values exceeding 0.75 in all cases. This suggests that the features provided strong discriminatory power for this class, even without transformation. On the other hand, Class 1 consistently showed the lowest performance, with AUC values below 0.66 across all configurations. This indicates a potential overlap in feature space or insufficiently informative features for distinguishing Class 1, highlighting the need for further feature engineering or the use of more advanced models.

Our results underscore the complex impact of feature transformation on classification performance. While transformation improved SVM performance, especially for Class 2, it reduced the effectiveness of MLR for Class 0 and Class 2. These findings suggest that the decision to apply feature transformation should consider the specific characteristics of the dataset, the target classes, and the model being used. For SVM, which often benefits from scaled or normalized data, feature transformation can enhance performance. However, for MLR, which relies on the linear relationships between features and target classes, transformation may obscure critical patterns.

Factors such as the quality and quantity of the training data, the choice of features used in the model, the complexity of the underlying relationships being modelled, and the tuning of hyperparameters may influence the accuracy levels of the MLR model in comparison to other machine learning classification algorithms. While MLR may not provide the same level of accuracy as more complex machine learning algorithms in some cases, it compensates with its high interpretability. Some of the machine learning algorithms that are often compared to MLR include decision trees, random forests, support vector machines, and neural networks. These algorithms are known for their ability to handle complex relationships and achieve higher accuracy in certain cases. However, they

often lack the interpretability that MLR offers, making MLR a valuable tool in situations where understanding the underlying relationships is essential.

5. Conclusions

A twelve-year period of data was considered in this study, and at the end of the first month, IPO returns were compared, and performance was determined. In order to determine the factors that significantly affect the performance of the companies on the Saudi stock market, the Multi Logistic Regression Model has been applied. Twelve financial variables, including Sector Code, Total Number of Offer Shares, Offer Price, Number of Substantial Shareholders, Subscription Quarter, Net Profit Margin, Return on Assets, Current Assets to Current Liabilities, Liabilities to Equity Ratio, Earnings per share, Individual Coverage, and Institutional Coverage, could be classified into three categories, BELOW AVERAGE, AVERAGE, or ABOVE AVERAGE. The prediction rate of 71.4 is good. Compared to similar studies on predicting stock market performance, a prediction rate of 71.4% is considered relatively high and indicative of a reliable model. This demonstrates the effectiveness of the Multi Logistic Regression Model in identifying the significant factors that impact company performance in the Saudi stock market.

Here, we include a detailed analysis of skewness, the application of transformations, and the evaluation of VIF values for the features, with certain features undergoing transformation. The skewness analysis and multicollinearity assessment confirms that all independent variables are well-suited for inclusion in the multinomial logistic regression model, as skewness corrections have normalized distributions and VIF values indicate no significant multicollinearity concerns. The comprehensive analysis highlights the critical role of feature transformations in improving the performance of the full MLR model, as evidenced by notable gains in accuracy and F1-scores, particularly in the AVERAGE category, where the model demonstrated consistent robustness. While the reduced model offers a simpler alternative, its lower accuracy and poorer recall for BELOW AVERAGE and ABOVE AVERAGE categories, especially when transformations are applied, underscore the trade-offs in predictive power. The model's superior performance across all metrics makes it the preferred choice for reliable classification, although the reduced model may be suitable for scenarios with resource constraints.

The transformation of features significantly improved the performance of the full MLR model, increasing overall accuracy from 71.43% to 79.60% and enhancing class-specific accuracy, particularly for the "BELOW AVERAGE" and "ABOVE AVERAGE" categories. However, the reduced model showed a decline in overall accuracy and a drop in the "BELOW AVERAGE" category's classification accuracy after transformation, suggesting that the reduced model struggled to generalize effectively with the transformed features. Feature transformations significantly impacted the parameter estimates in both the full and reduced MLR models, with transformations leading to reduced statistical significance and increased standard errors in the full model, while the reduced model retained the significance of key variables like SQ, NPM, and LER, showing improved robustness. These findings underscore the importance of evaluating transformations carefully, as they can alter the interpretability and reliability of regression models by influencing coefficient magnitudes and significance levels.

The analysis underscores the importance of variable importance rankings in predictive modeling, highlighting that features like SQ, SC, and NPM% consistently play a significant role in predicting IPO performance, even after transformations. The observed shifts in feature importance due to transformations, such as the enhanced relevance of `ins_cov%` and the diminished impact of EPS,

emphasize the dual role of feature engineering in improving model accuracy while potentially altering the interpretability of individual predictors, necessitating a balanced approach to preprocessing. The comparison of within-sample metrics, including precision, recall, and F1 scores, with out-of-sample validation accuracy for both full and reduced MLR models reveals that while feature transformations improved within-sample metrics, they caused a significant drop in validation accuracy, indicating challenges in generalization. This finding highlights the importance of a balanced approach to feature engineering and model complexity, as non-transformed features demonstrated greater robustness and reliability for the given dataset and context.

In conclusion, the comparative analysis highlights the strength of the Multinomial Logistic Regression (MLR) model in achieving the highest overall accuracy (71.4%) and balanced performance across categories, with notable results in terms of AUC values. For untransformed features, MLR outperformed Support Vector Machine (SVM) in Class 0 (AUC 0.86 vs. 0.61) and Class 1 (AUC 0.66 vs. 0.58), while both models performed comparably for Class 2 (AUC 0.81 for MLR vs. 0.86 for SVM). After feature transformation, SVM excelled in Class 2, achieving an AUC of 0.91 compared to MLR's 0.75, but MLR retained higher AUC values for Class 0 (0.71 vs. 0.63) and Class 1 (0.64 vs. 0.59). These results emphasize the nuanced impact of feature transformation: While it enhanced SVM's performance, especially for Class 2, it diminished MLR's ability to discriminate among certain classes. Overall, MLR's capacity to consistently deliver strong AUC values for most classes underscores its reliability and suitability for datasets with complex categorical relationships, balancing accuracy, interpretability, and robust performance across diverse conditions. Overall, feature transformations improved the full MLR model's performance within the training set, enhancing accuracy and F1-scores, particularly for "BELOW AVERAGE" and "ABOVE AVERAGE" categories; however, they introduced challenges in generalization, leading to reduced validation accuracy and altered parameter estimates. Despite these limitations, the MLR model consistently demonstrated superior reliability and balanced performance across categories, making it a robust choice for datasets with complex categorical relationships.

Author contributions

M.F.A.: Conceptualization, Methodology, Data Curation, Review & Editing.

M.T.Y.: Software, Machine Learning analysis, Writing - Original Draft, Writing.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Conflict of interest

The authors declare no conflict of interest.

Reference

- Ali SS, Mubeen M, Hussain A (2018) Prediction of stock performance by using logistic regression model: Evidence from Pakistan Stock Exchange (PSX). In: *Patron of the Conference*, 15. <https://doi.org/10.18488/journal.1007/2018.8.7/1007.7.247.258>
- Anand E, Pandya G (2023) Prediction of IPO subscription—A logistic regression model. *SDMIMD J Manage* 14. Available from: <https://informaticsjournals.co.in/index.php/sdmimd/article/view/33253>.

- Ashford K (2023) *What is an IPO?* Forbes. Available from: <https://www.forbes.com/advisor/investing/initial-public-offering-ipo/>.
- Baba B, Sevil G (2020) Predicting IPO initial returns using random forest. *Borsa Istanbul Rev* 20: 13–23. <https://doi.org/10.1016/j.bir.2019.08.001>
- Connelly B, Certo S, Ireland R, et al. (2011) Signaling theory: A review and assessment. *J Manage* 37: 39–67. <https://doi.org/10.1177/0149206310388419>
- Daily CM, Certo ST, Dalton DR, et al. (2003) IPO underpricing: A meta-analysis and research synthesis. *Entrep Theory Pract* 27: 271–295. <https://doi.org/10.1111/1540-8520.t01-1-00>
- Das N, Sadhukhan B, Chatterjee T, et al. (2022) Effect of public sentiment on stock market movement prediction during the COVID-19 outbreak. *Soc Netw Anal Min* 12: 92. <https://link.springer.com/article/10.1007/s13278-022-00919-3>
- DeCastro BR (2019) Cumulative ROC curves for discriminating three or more ordinal outcomes with cutpoints on a shared continuous measurement scale. *Plos One* 14: e0221433. <https://doi.org/10.1371/journal.pone.0221433>
- Emidi C, Galán S (2022) Prospectus content as predictor of IPO outcome: A topic model approach. *J Behav Exp Financ* 34: 100669. Available from: <https://lup.lub.lu.se/student-papers/search/publication/9083567>.
- Feng CH, Disis ML, Cheng C, et al. (2022) Multimetric feature selection for analyzing multicategory outcomes of colorectal cancer: Random forest and multinomial logistic regression models. *Lab Invest* 102: 236–244. <https://doi.org/10.1038/s41374-021-00662-x>
- Hanbing Z, Jarrett JE, Pan X (2019) The post-IPO performance in the PRC. *Int J Bus Manage* 14: 109–138. <http://doi.org/10.5539/ijbm.v14n11p109>
- Hosmer Jr DW, Lemeshow S, Sturdivant RX (2013) *Applied logistic regression* (3rd ed.). Wiley. <https://doi.org/10.1002/9781118548387>
- Huang S, Mao Y, Wang C, et al. (2021) Public market players in the private world: Implications for the going-public process. *Rev Financ Stud* 34: 2411–2447. <https://doi.org/10.1093/rfs/hhaa092>
- Kagan J (2023) *Underpricing: Definition, how it works, and why it's used*. Investopedia. Available from: <https://www.investopedia.com/terms/u/underpricing.asp>.
- Kooli M, Meknassi S (2007) The survival profile of US IPO issuers: 1985–2005. *J Wealth Manage* 10: 105–119. Available from: <https://www.pm-research.com/content/ijwealthmgmt/10/2/105>.
- Kraus M, Feuerriegel S (2017) Decision support from financial disclosures with deep neural networks and transfer learning. *Decis Support Syst* 104: 38–48. <https://arxiv.org/abs/1710.03954>
- Loughran T, McDonald B (2011) When is a liability not a liability? Textual analysis, dictionaries, and 10-Ks. *J Financ* 66: 35–65. <https://doi.org/10.1111/j.1540-6261.2010.01625.x>
- Multinomial logistic regression. (2023) *Bookdown*. Available from: https://bookdown.org/chua/ber642_advanced_regression/multinomial-logistic-regression.html.
- Multinomial logistic regression (2023) *UCLA Institute for Digital Research and Education*. Available from: <https://stats.oarc.ucla.edu/stata/dae/multinomiallogistic-regression/>.
- Nguyen TH, Shirai K, Velcin J (2015) Sentiment analysis on social media for stock movement prediction. *Expert Syst Appl* 42: 9603–9611. <https://doi.org/10.1016/j.eswa.2015.07.052>
- Ni S (2023) Predicting IPO performance from prospectus sentiment. *BCP Bus Manage* 38: 3063–3075. <https://doi.org/10.54691/bcpbm.v38i.4237>
- Pedregosa F, Varoquaux G, Gramfort A, et al. (2011) Scikit-learn: Machine learning in Python. *J Mach Learn Res* 12: 2825–2830. Available from: <https://jmlr.org/papers/v12/pedregosa11a.html>.
- Qutab I, Malik KI, Arooj H (2022) Sentiment classification using multinomial logistic regression on roman Urdu text. *Int J Innov Sci Technol* 4: 223–335. <https://doi.org/10.33411/IJIST/2022040204>

- Ranganathan P, Pramesh C, Aggarwal R (2017) Common pitfalls in statistical analysis: Logistic regression. *Perspect Clin Res* 8: 148–151. https://doi.org/10.4103/picr.PICR_87_17
- Schumaker RP, Chen H (2009) Textual analysis of stock market prediction using breaking financial news: The AZFin text system. *ACM T Inform Syst* 27: 1–19. <https://doi.org/10.1145/1462198.1462204>
- Schumaker RP, Zhang Y, Huang CN, et al. (2012) Evaluating sentiment in financial news articles. *Decis Support Syst* 53: 458–464. <https://doi.org/10.1016/j.dss.2012.03.001>
- Tao J, Deokar AV, Deshmukh A (2018) Analysing forward-looking statements in initial public offering prospectuses: A text analytics approach. *J Bus Anal* 1: 54–70. <https://doi.org/10.1080/2573234X.2018.1507604>
- Upadhyay A, Bandyopadhyay G, Dutta A (2012) Forecasting stock performance in Indian market using multinomial logistic regression. *J Bus Stud Q* 3: 16–28.
- Vryniotis V (2023) *Machine learning tutorial: The multinomial logistic regression (softmax regression)*. Datumbox. Available from: <https://blog.datumbox.com/machine-learning-tutorial-the-multinomial-logistic-regression-softmax-regression/>.



AIMS Press

© 2025 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<https://creativecommons.org/licenses/by/4.0>)