Akademia Górniczo-Hutnicza im. Stanisława Staszica w Krakowie

Wydział Informatyki

Praca dyplomowa

# Comparison of selected machine learning methods for prediction and detection of outliers used in the study of the impact of changes in the capital and property structure of a company on its market value

**Porównanie wybranych metod uczenia maszynowego przeznaczonych do predykcji i wykrywania wartości odstających zastosowanych w badaniu wpływu zmian struktury kapitałowej i majątkowej przedsiębiorstwa na jego wartość rynkową**

| | |
|---|---|
| Autor: | **Jakub Stanisław Janicki** |
| Kierunek: | **Informatyka** |
| Opiekun pracy: | **dr hab. inż. Rafał Dreżewski, prof. AGH** |

Kraków, 2024

## Abstract

The thesis is based on an analysis of the hypothesis, which states: the capital and property structure of a company has an impact on its value. The claim will be tested using data that every joint-stock company in Poland is required to make available.

The data from numerous companies, spanning a multitude of years and representing an industry that undergoes constant change, may contain a considerable number of anomalies. To identify and eliminate these anomalies, a number of fundamental machine learning techniques were employed, including Local Outlier Factor, Isolation Forest, and One-Class Support Vector Machine. Additionally, a neural network with an autoencoder architecture was employed. The efficacy of these algorithms was examined visually through the use of t-SNE and PCA methods. The quality of the prediction was also evaluated on the dataset before and after the removal of anomalies.

The hypothesis analysis employed basic machine learning methods, including K-Nearest Neighbours, Decision Trees, Support Vector Machines, and Logistic Regression. Additionally, Deep learning methods, such as deep Neural Networks and Convolutional Neural Networks, were utilised. Prior to prediction, the data underwent scaling, transformation, and cleaning. For the Convolutional Network, Simulated Annealing was used in the transformation of numerical data to images. This produces images optimised with respect to the mutual distance and correlation of the variables. The problem is presented as both classification and prediction, and the models are compared with each other using appropriate metrics.

The experiments demonstrate that the removal of the anomaly was performed correctly. This action significantly improved the quality of the prediction. The results indicate that enterprise value regression is too complex a task. Consequently, we presented the problem as a classification. Upon analysis of the experiments, it was determined that they did not provide sufficient evidence to confirm the hypothesis.

## Streszczenie

Praca opiera się na analizie hipotezy, która brzmi: struktura kapitałowo-majątkowa przedsiębiorstwa ma wpływ na jego wartość rynkową. Zostanie ona zweryfikowana z wykorzystaniem danych, które każda spółka akcyjna w Polsce ma obowiązek udostępnić.

Dane pochodzące z wielu firm, z szerokiego zakresu lat, z branży, która dynamicznie się zmienia, mogą zawierać znaczną liczbę anomalii. Aby je wykryć i usunąć, wykorzystano szereg podstawowych metod machine learningu, takich jak Local Outlier Factor, Isolation Forest, One-Class Support Vector Machine. Użyto także sieci neuronowej o architekturze autoenkodera. Działanie tych algorytmów sprawdzono wizualnie metodami t-SNE oraz PCA. Sprawdzono również jakość predykcji na zbiorze danych przed i po usunięciu anomalii.

W analizie hipotezy wykorzystano podstawowe metody machine learningu, takie jak K-Nearest Neighbours, Decision Tree, Support Vector Machine, Logistic Regression. Użyto również metod deep learningu, takich jak Deep Neural Network oraz Convolutional Neural Network. Przed przystąpieniem do predykcji dane są odpowiednio skalowane, transformowane oraz czyszczone. Dla sieci konwolucyjnej, w transformacji danych numerycznych do obrazów, wykorzystuje się metodę Simulated Annealing. W ten sposób otrzymujemy obrazy zoptymalizowane względem wzajemnej odległości i skorelowania zmiennych. Problem został przedstawiony zarówno jako klasyfikacja, jak i predykcja, a modele porównano ze sobą za pomocą odpowiednich metryk.

Z przeprowadzonych eksperymentów wynika, że usunięcie anomalii wykonano prawidłowo. Działanie to znacznie poprawiło jakość predykcji. Rezultaty pokazują, że regresja wartości przedsiębiorstwa jest zbyt złożonym zadaniem. Z tego powodu przedstawiliśmy problem jako klasyfikację. Po analizie eksperymentów stwierdzono, że nie pozwalają one na potwierdzenie hipotezy.

# Contents

# List of Figures

# List of Tables

# List of Algorithms

# 1 Introduction

Although Machine Learning concepts were introduced in the early fifties, the first paper describing the application of ML methodology to an economics problem was published in 1984 [1]. This thesis presented an approach to the issue of daily returns on IBM common stock. The author used neural networks, which were not sufficiently developed at the time, resulting in experiment failure [2]. Currently, the situation has changed.

Economics is a field where datasets can be classified as big data. This is due to the substantial growth of economics as a vital branch of contemporary science. The significance of this discipline is demonstrated by the award of the Economics Nobel Prize, which is one of the four major science field Nobel Prizes. This highlights the importance of economics in present–day society. Given the desire to analyse and enhance past records, there is a necessity for data storage. Every stock exchange provides clients with historical data on market behavior, which is both accurately stored and publicly available. As a demonstration, we can reference the Giełda Papierów Wartościowych w Warszawie, which has stored and shared this type of data since 1998 [3].

Currently, Machine Learning has progressed significantly, and over the past two decades, there has been a great deal of growth in ML and neural networks. This technology has enabled us to create autonomous cars, and robots, diagnose diseases, and much more [4]. Deep Learning is now utilized in image recognition, natural language processing, and solving differential equations. Consequently, in 2020, OpenAI released the GPT–3 tool, which has made AI accessible to the public.

As the problems from the beginning were mainly solved, scientists started to use ML in every Economic problem. The Special Issue which consists of state–of–the–art usages of ML in finances was composed of 17 papers that play a major role in providing insight in Economics. Some of the papers covered such topics as the nowcast of the GDP growth, identifying determinants of the Bitcoin price, forecasting shipping prices, forecasting unemployment rate, forecasting the price of gold, etc. [1].

One of the prevalent economic issues that remains unsolved is forecasting an optimal business strategy. This strategy entails balancing resources between assets and liabilities, known as capital–property structure. Modigliani–Miller Theory posits that this structure has no impact on a company's market value [5]. Nonetheless, the assumptions of this theorem are unrealistic as it assumes an idealistic market. This model does not account for taxes, commissions, or stock exchange schemes [6]. However, in reality, several determinants need consideration when analysing this structure. Defining precise equations that determine the optimal solution is challenging [7].

Moreover, due to the ubiquity of this issue, there is a vast amount of data available. Companies listed on the stock market regularly generate records that exhibit significant variance in content. These organizations operate across different sectors, locations, and scales, necessitating careful consideration of data accuracy before any analysis. Using inaccurate data for training models could result in failures. Therefore, when dealing with large and authentic datasets, data selection and preprocessing pose significant challenges.

In summary, the objective of this thesis is to analyse the hypothesis, which states: The capital and property structure of a company has an impact on its value.

Exploring this subject is significant as the financial market is characterised by fluctuations and competition. ML methods offer an advantage in managing this issue. Addressing this problem can lead to the proper adjustment of companies to market needs, potentially increasing market value. Data preprocessing is also valuable in this dissertation. Preparing the data by removing outliers is an investment that can benefit everyone who works with the data and significantly accelerate research progress.

Taking into account the aforementioned issues, this thesis will define the scope of Machine Learning as a solution.

Firstly, proper data preparation is crucial, which involves addressing outliers [8]. A comparison of selected machine learning techniques will be employed to identify and eliminate the outliers, thus enhancing the accuracy of the model that can be customized for specific concerns.

Subsequently, the selected ML techniques will be employed to forecast the market value of the corporation in relation to its capital and property configuration.

An elaboration of the goals outlined will be presented in the subsequent chapters.

1. Chapter one, “Review of Related Research”, provides an overview of the research on Machine Learning in Economics. The analysis will primarily focus on the aspects of outliers and prediction.

2. Chapter two, “Problem Formulation and Proposed Solution”, describes the problem from both the economics and machine learning perspectives. The chapter then introduces a set of algorithms and explains their suitability for addressing the problem. Additionally, it will detail the selection of metrics that will be used to compare the solutions. Practical solutions, including code snippets and implementation details, will supplement the theoretical discussion.

3. Chapter three, “Experimental Results”, presents the findings of the experiments conducted. A comparative analysis of the utilized algorithms will be displayed, accompanied by the use of chosen metrics.

4. Final section is“Conclusions”. This section analyses the findings and approach taken to address the problem, as well as provides proposals for future project development.

# 2 Review of Related Research

This chapter explores three important issues. Firstly, it examines the current state of knowledge in solving economic problems with machine learning. Secondly, it focuses on methods for detecting outliers in these issues. Finally, it outlines the most valuable approaches for predicting data. State–of–the–art solutions presented in cited papers are provided for each of these issues.

## 2.1 Fundamental concepts

### 2.1.1 Capital–property structure

This balance is a statement of assets and liabilities, that companies are supposed to draw up at the beginning and end of the reporting period [9].

1. Assets – what the company has. They can be divided based on how long they remain within a company:
   - Fixed assets – are resources that remain in the company for more than 12 months from the balance sheet date. They are characterized by a low degree of liquidity, which means that they cannot be converted into cash in a short time. For instance: property, plant and equity, intangible assets, non–current loans, and receivables.
   - Current assets – are intended for sale. They may be materials intended for own consumption, manufactured or processed finished products, semi–finished products, and goods purchased for resale.
2. Equity & Liabilities – ways in which we obtain funds to finance our assets. Liabilities are divided into:
   - Equity – is not uniform – its structures besides all the others include two main parts. Basic (share) capital – is the owners' contribution to the company's assets, which allows it to start operations. Second is supplementary capital – may arise from the surplus of the price over the nominal value of shares or during the company's operation when its supplementary capital is increased.
   - Liabilities – external capital includes capital placed at the company's disposal for a specified period, after which it should be returned. Until foreign capital is

returned, it remains a liability. It can be divided into long–term and short–term liabilities with a repayment period of up to one year. Foreign capital, for instance, may be non–current loans and borrowings, or current liabilities from derivatives.

### 2.1.2 Market value

Market value is the price an asset would fetch in the market, based on the price that buyers are willing to pay and sellers are willing to accept. It is usually calculated by multiplying the number of outstanding shares by the current share price. Market value is dependent on numerous other factors, such as the sector in which the company operates, its profitability, debt load, and the broad market environment [10].

## 2.2 Economic problem

Machine learning has gained popularity in economics. However, we have limited our research to basic terms such as *'capital structure'* and *'balance sheet'* due to the complexity of our data. Through this approach, we have identified similar issues to our own and presented them (Fig. 2.1).

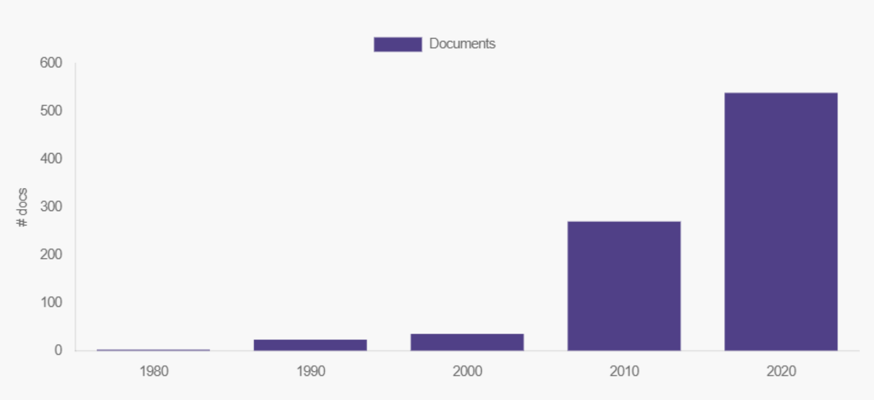

Figure 2.1: Number of economics publications containing the base words *'capital structure'*, *'balance sheet'*, and *'machine learning'* [11]

This thesis topic has been extensively researched. One researcher analysed data from 34 companies quoted on the Ghana Stock Exchange for the year ended 31st December 2010, focusing solely on the long–term debt and equity fields of the capital structure. The choice was justified by the publishers. Equity can be defined as retained earnings that are reinvested into a company to increase its value. On the other hand, long–term debt can have both positive and negative effects on a firm's value, as it may lead to under or over–investment problems.

The author explores these two theses using the Ordinary Least Square (OLS) technique. The model prepared elucidates the relationship between the dependent variable (firm value) and the independent variable. The R–Bar–squared metric was equal to 1, indicating a strong correlation between the variables. The Standard Error of Regression and F–calculate values suggest relatively good prediction. Upon analysis of the results, the author concludes that long–term debt and equity are positively correlated with the market value of firms. It is important to note that this paper only describes a part of the capital structure, not the property structure [12].

Another approach to the problem considered 8,459 records collected from 769 companies in the Vietnamese stock market in 2012/22. The study distinguished three dependent variables to measure firm value.

- ROA – indicates how effectively a company utilizes its assets to generate profit. It is represented by equation (Eq. 2.1).

$$ROA = \frac{\text{Net Income}}{\text{Total Assets}} \tag{2.1}$$

- ROE – represents the return generated for each dollar of shareholders' equity invested in the company. It is represented by equation (Eq. 2.2).

$$ROE = \frac{\text{Net Income}}{\text{Shareholders' Equity}} \tag{2.2}$$

- Tobin's Q – compares the market value of a company's assets to the replacement cost of those assets. It is represented by equation (Eq. 2.3).

$$Tobin's\ Q = \frac{\text{Market Value of Firm's Assets}}{\text{Replacement Cost of Firm's Assets}} \tag{2.3}$$

The model's independent variables are not limited to pure capital structure fields but also include economic measures such as debt–to–assets (Lia), long–term debt–to–assets (Llia), and short–term and long–term debt–to–assets (Tlia), as well as firm size. The authors applied various regression methods, including OLS, FEM, REM, and GLS. The results of the GLS model suggest that Lia has a positive influence on all three value indicators, with the strongest impact on Tobin's Q. The long–term debt ratio, on the other hand, has no impact on firm value. The use of short–term and long–term debt ratios has a negative influence on all three metrics used to measure firm value. Therefore, executives should rely less on short–term debt and explore alternative capital sources [13].

A study was conducted on data from 55 companies listed on the Tehran Stock Exchange between 2010 and 2014. Multiple regression analysis was used to test hypotheses and evaluate the significance of various factors, with a 95% confidence level for both F–statistics and t–tests. The results show that the return on equity (ROE) has a statistically significant negative impact on financial leverage. Furthermore, the financial leverage of the company is positively impacted by the market value of its earnings per share (EPS). The coefficient associated with firm size also shows a significant positive association with the company. However, the analysis indicates that growth does not have a significant effect on firm value, as the significance level of 5% suggests [14].

Basic statistical methods are commonly used to analyse the influence of capital structure on company value. However, it is important to note that the methods and results can vary significantly, indicating a strong dependence on industry or country. It should be noted that unlike the authors mentioned above, we do not take into consideration the balance sheet.

## 2.3 Anomaly detection

To begin, we must first define what an anomaly is.

> *Rare items, events, or observations that deviate significantly from the majority of the data and do not conform to a well–defined notion of normal behavior. Such examples may arouse suspicions of being generated by a different mechanism, or appear inconsistent with the remainder of that set of data [15].*

Since economics is strongly associated with money, As business is closely linked to money, there is no doubt that fraud has become more prevalent as the economy continues to grow. Fraudsters are constantly evolving their approaches to exploit vulnerabilities in the financial sector. Anomalies can be caused not only by suspicious activity, but also by external conditions such as system failures, network outages, and so on. Therefore, anomalies can occur in all sectors of the economy. The Crime Complaint Center reports over 400,000 complaints of internet crime in the US, where a total of over 3.5 billion $ was lost, an increase of 30% from the previous year (2018) [16]. The most common are money laundering, insurance fraud, and credit card fraud.

Once the significance of the problem has been defined, solutions need to be worked out. The approach to a problem depends heavily on the completeness of the data. The availability of labeled data is a crucial factor in the process of selecting a method. Depending on this, we can choose one of the supervised, semi–supervised, or unsupervised methods [17].

Having our case specification (Tab. 2.1), we will only focus on unsupervised anomaly detection methods.

### 2.3.1 Isolation Forest

First introduced in 2008, it is based on binary divisions, resulting in linear time complexity and low memory consumption, so it is often chosen for larger datasets. In 2018, a solution was proposed to detect fraudulent credit card transactions. The dataset consists of about 300,000 transactions with only a few hundred fraud records. It was compared with OCSVM, LOF, and k–means algorithms. It had the highest accuracy with an AUC of 95.12% [18].

Another paper that used IF to detect outliers in economic issues was created based on 9500 samples of workers' compensation claims. Numerous ML models were tested such as LR, DT, RF, Linear Kernel SVM, and Radial Basis Function Kernel SVM to predict the label of the record. The AUC of each model was compared with and without anomalies detected by the IF method. The highest AUC achieved by the linear SVM was 87.72% without anomalies, compared to the linear SVM with anomalies, which scored about 3–5% less [19].

Table 2.1: Anomaly Detection Methods Categorized [17]

| Type | Methods |
|---|---|
| Supervised | Multilayer Perception (MLP)<br>Convolutional Neural Networks (CNN)<br>Long Short–Term Memory Networks (LSTM)<br>Naive Bayes (NB) |
| Unsupervised | Isolation Forest (IF)<br>Autoencoders (AE)<br>Local outlier factor (LOF)<br>One–Class Support Vector Machines (OCSVM)<br>k–means Clustering (k–MC)<br>Density–Based Spatial Clustering of Applications with Noise (DBSCAN)<br>k–Nearest neighbours (k–NN)<br>Decision Trees (DT)<br>Random Forest (RF) |
| Semi–Supervised | Hidden Markov Models (HMM)<br>Generative Adversarial Networks (GAN)<br>GA and unsupervised FCM clustering (GAFCM) |

### 2.3.2 Autoencoders

Autoencoders were employed to identify fraudulent credit card transactions in a dataset comprising records from 1000 German credit cards. The authors proposed training an AE and adding a softmax layer to classify the output. Two AE configurations were tested. The first is with 20 neurons in the input layer, then 15, 10, and 5 neurons in the next hidden layers of the first symmetrical half of the network. The second with 20, 30, 50, and 100 neurons each in the following layers. The test showed that both models performed satisfactorily. However, AE with a more extensive architecture had a better performance with an accuracy of 84.1% [20].

The architecture of autoencoders can vary, as can the way they are used. Pumsiriart and Yan in their research created AE with 21 input neurons, 16.8 and 4 neurons in each layer of the first half of the structure with tangent activation functions and MSE as a measure of reconstruction error. Before applying the AE, they parsed the data with PCA transformation. The data consists of 3 datasets, all consisting of credit card transactions. They differ in size and location of data collection. The results were presented with a comparison to the RBM model. On the dataset from Germany, AE achieved an AUC of 43.76% while RBM achieved 45.62%. On the dataset from Australia, AE achieved an AUC of 54.83% compared to 52.38% for RBM. A significant jump in performance was observed on the dataset from Europe, which was significantly larger than the others. AE achieved an AUC of 96.03% while RBM achieved

95.05%. Therefore, it could be concluded that both models had satisfactory performance in detecting fraudulent credit card transactions on large datasets as opposed to small datasets [21].

### 2.3.3 Local outliers factor

Calculates a metric for each dataset, which in turn indicates the sparseness of a data point concerning its k nearest neighbours. In this way, it accounts for local density variations between data points. Observations with a high LOF value are considered as anomalies. This method was first proposed by Markus M. Breunig, Hans–Peter Kriegel, Raymond T. Ng, and Jörg Sander in 2000 [22]. Since then, extensions and modifications of LOF have been proposed. Some of them are based on nearest neighbour taxonomy like COF[23], LOCI [24], aLOCI [24], INFLO [25], LoOP [26]. Others are based on clustering taxonomy like LDCOF [27], CBLOF [28]. We can choose between them based on the size and characteristics of the input data [29].

In 2003, researchers used LOF to identify various network intrusions. LOF was compared with several supervised and unsupervised prediction methods. The solutions were evaluated on the DARPA 1998 dataset of network connections as well as on real network data. The results were promising: LOF was able to achieve detection rates of 74% and 56% for various attacks while keeping the false alarm rate at 2%. When the rate was increased to 4%, the rates reached 89% for bursty attacks and 100% for single connection attacks. This outperforms other methods such as Neural Networks, Mahalanobis method, and Unsupervised SVM [30].

### 2.3.4 One–class Support Vector Machine

Although SVM is a computationally intensive method, it can be applied to high–dimensional and large datasets. The problem is that the presence of irrelevant features can mask the presence of anomalies. This problem is commonly known as the 'curse of dimensionality'. Therefore, when building an SVM model for anomaly detection, we sometimes need to combine it with an unsupervised feature extractor. Thus, researchers propose an architecture that is a hybrid model where an unsupervised DBN is trained to extract generic anomaly features, while a one–class SVM is trained from the features learned by the DBN. Having known that linear kernels can be substituted for nonlinear ones in this hybrid model, without loss of quality. The results show that this hybrid model is comparable to deep AE and faster. The time for testing and training was reduced by 1000 and 3 times respectively. The research was performed on a dataset of logs from IoT devices, where the aim was to detect unusual behavior caused by either faulty devices or events of interest in the monitoring environment [31].

## 2.4 Predictions

After filtering anomalies from our data, we can proceed to make predictions. In the field of economics, there are several papers available. Most of them address the common issue of predicting bankruptcy using financial statements.

### 2.4.1 Convolutional Neural Networks

The use of convolutional neural networks in financial analysis has only been reported in a small number of studies. This is because they are more suitable for image analysis and less suitable for general numerical data, including financial statements. However, they have been used as an approach for bankruptcy prediction, which is a two–class classification problem. The research uses balance sheets and profit–and–loss statements from approximately two thousand companies listed on the Japanese stock market, resulting in 7520 records. The data is represented as greyscale images using two methods: random and correlated. In the random method, pixels are assigned traits randomly, while in the correlated method, the correspondence between financial ratios and pixel positions is determined so that highly correlated financial ratios are placed close to each other. Thus, CNN was able to recognize extremely large correlations between neighbouring pixels. The two methodologies were compared using a convolutional neural network based on GoogLeNet. It was found that the correlated method was more appropriate for this purpose. This analysis also demonstrated that their models outperformed other machine learning methods such as CART, LDA, SVM, MLP, AdaBoost, and Altman's Z–score. Their model achieved an accuracy rate of over 90% [32]. A comparison of ML models considered in this paper was visualised (Fig. 2.2).

### 2.4.2 Multilayer Perceptron

The Multilayer Perceptron (MLP) is a contemporary feedforward neural network comprising fully connected neurons and typically employing nonlinear activation functions such as ReLu or sigmoid. It consists of input and output layers, as well as one or more hidden layers with numerous stacked neurons.The backpropagation algorithm is used to minimise the cost function.

This thesis presents a solution for bankruptcy prediction using MLP. The dataset consists of balance sheets and profit–and–loss statements from industrial companies operating in the Czech Republic over the past five years. The author compares MLP with different configuration properties, including varying the number of neurons in each layer, as well as the error function, hidden activation function, and output activation function. A comparison of MLP different variants addressed in this paper was visualised (Fig. 2.3).

The network configuration with 22 neurons in the input layer, 6 in the hidden layer, and 2 in the output layer outperformed other configurations, achieving an accuracy score of 83%.

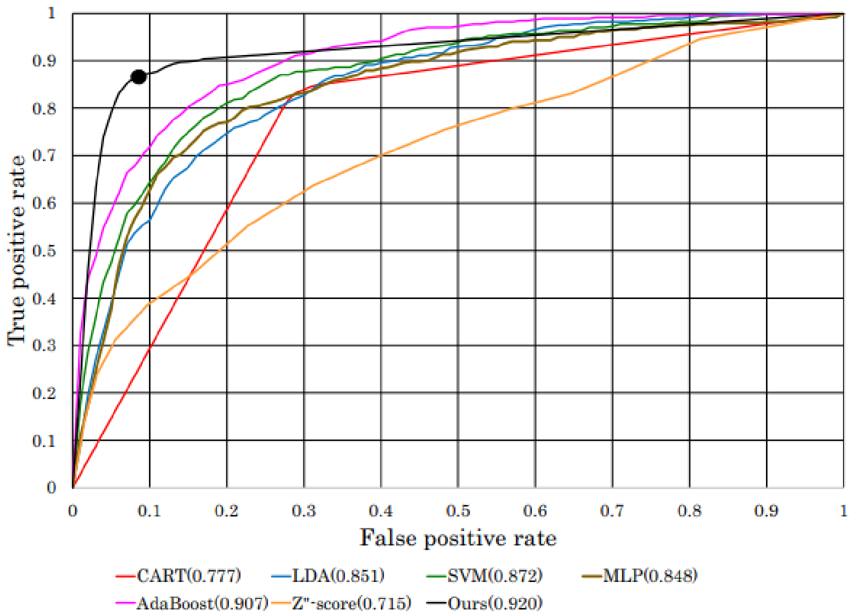

Figure 2.2: ROC curve for ML methods comparison with CNN [32]

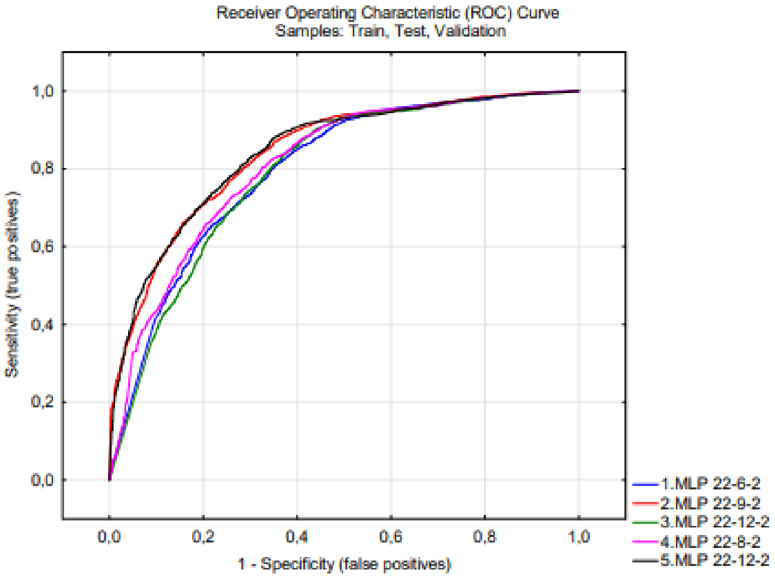

Figure 2.3: Comparison of MLP different variants [33]

This best–performing network was also compared to an SVM, which achieved an accuracy score of 76% [33].

### 2.4.3 Long short–term memory neural network

LSMT is a type of recurrent neural network that aims to solve the vanishing gradient problem present in traditional RNNs. The LSTM model is composed of a memory cell that maintains its state over time and three gates: forget, input, and output. The forget gate decides whether to keep or forget information from the previous time step, while the input gate quantifies the importance of new information carried by the input. The output gates manage the information in the current state and output it, taking into account both the previous and current states.

Again, the process of predicting bankruptcy is considered. The author predicts the future development of a company operating in the manufacturing sector in the Czech Republic. The architecture of LSTMs could vary significantly, so many variations are created and compared. The researchers modulate the size of the hidden layers and their activation functions. Neural network with the best configuration has excellent results. Out of 487 active companies, the neural network was able to identify 473 companies able to survive potential financial distress; out of 262 companies identified as going bankrupt, the neural network identified 182. In this case, the network has 14 neurons in the input layer and 940 in the LSTM layer. In both elementary hidden layers, it was a function of the hyperbolic tangent [34].

Many methods are used together with LSTM. In this paper, the author decided to improve the LSTM model by using Empirical Mode Decomposition (EMD) and Complete Ensemble Empirical Mode Decomposition with Adaptive Noise (CEEMDAN). First the data is decomposed with EMD and CEEMDAN, then this is used for prediction with LSTM, and then finally is reconstructed to obtain the final value. Comparison with SVM, WAV was done on two datasets. They consist of American and Chinese stock price datasets. Subsequently, the model with LSTM–CEEMDAN obtained the best results in terms of errors. The prediction errors are smaller, ranging from a few to several hundred percent [35].

### 2.4.4 Other ML methods

So far, our research has concentrated on deep learning, which is currently the most advanced field in this area. However, other ML solutions are also valuable for this type of problem. The researchers were able to compare all the most popular ML methods. They used 13 financial information items for about 1020 companies listed on KOSPI and KOSDAQ from 2012 to 2021 and compared the bankruptcy prediction performance of LSTM, LR, k–NN, DT, and RF models. It was found that LSTM is not a suitable model due to the low number of bankruptcies. The rest of these methods were tested and compared with the ROC AUC metric. The results are satisfactory, nevertheless cross validation techniques were used because of the small data size [36]. A comparison of ML models considered in this paper was visualised (Fig. 2.4).

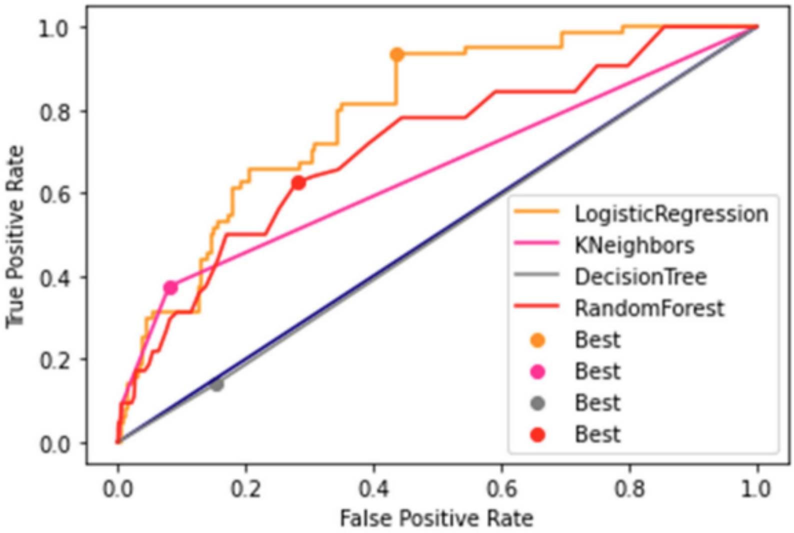

Figure 2.4: Comparison of ML methods [36]

# 3 Problem Formulation and Proposed Solution

This chapter outlines the characteristics of the data and its analysis. Data processing will consist of three main parts: data preprocessing, outlier detection, and market value prediction. The figure below (Fig. 3.1) illustrates this process. Each section proposes and describes representative solutions.

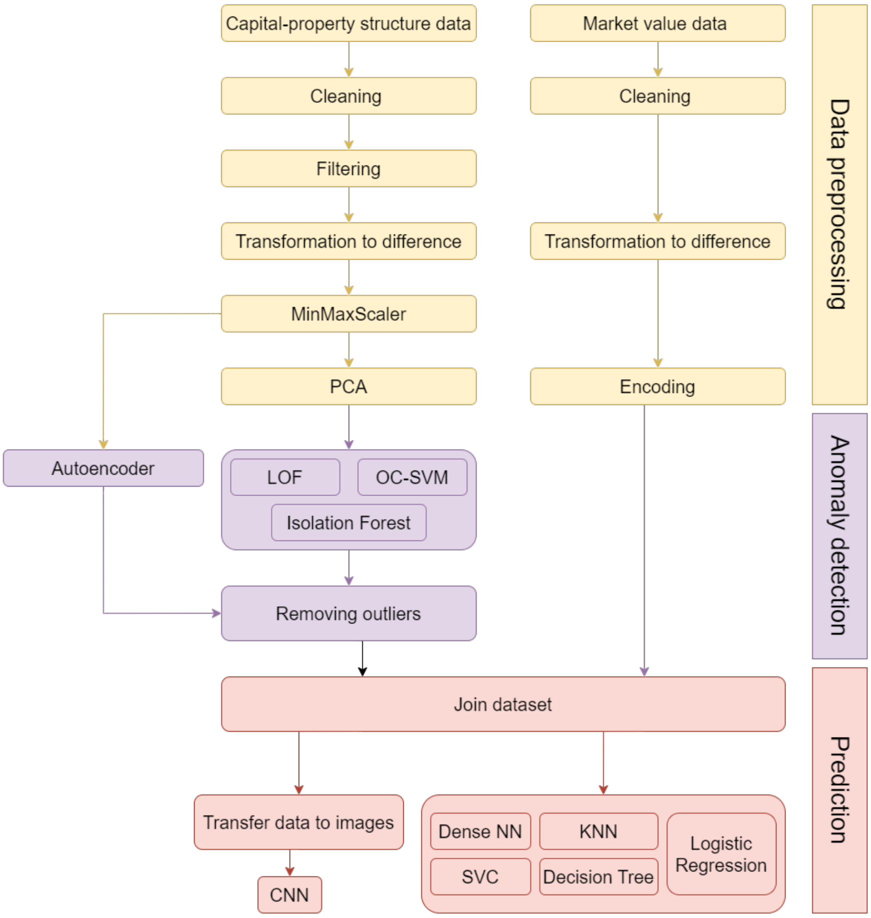

Figure 3.1: Workflow described in third chapter

## 3.1 Dataset details

### Capital–property structure

The dataset comprises 270 Excel files, each pertaining to a specific company. In addition to the balance sheet, each sheet contains metadata such as the company name, sector, and owner. The sheets represent reports drawn up over various years, with the number of sheets varying between companies due to bankruptcy or subsequent firm establishment. The issue is illustrated in figure (Fig. 3.2), which presents the number of records per company from the 'IT Systems' sector.

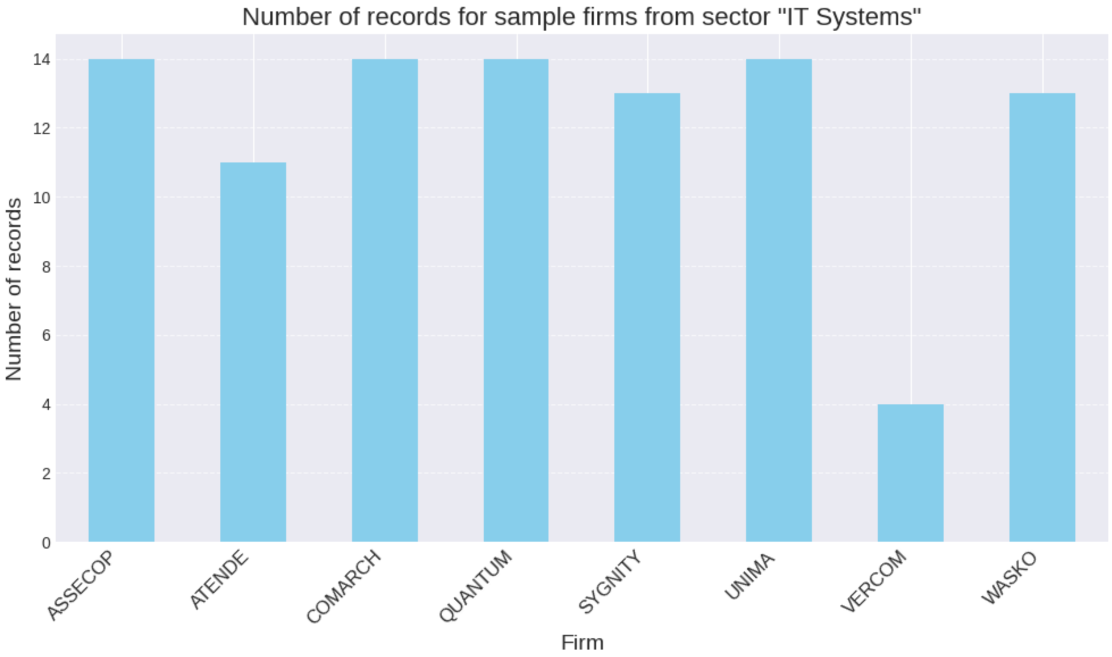

Figure 3.2: Number of capital–property structure records for sample companies

For instance, Żywiec company published a report between 1997 and 2021, with one such report presented in (Tab. 3.1). The structure presented in the example may have empty values, which can vary depending on the company sector. It is important to note that this structure is unique to each use.

Table 3.1: Balance sheet for Żywiec company for the years 1997–98. The presented values are in PLN .

| **End of period** | 1997–12–31 | 1998–12–31 |
|---|---|---|

| | | |
|---|---|---|
| **ASSETS** | 402,760 | 1,755,951 |
| Non–current assets | 312,860 | 1,398,742 |
| Property, plant and equipment | 292,092 | 1,334,765 |
| Exploration for and evaluation of mineral resources | | |
| Intangible assets | 2,160 | 31,981 |
| Goodwill | 5,397 | |
| Investment property | | |
| Right–of–use assets | | |
| Investment in affiliates | | |
| Non–current financial assets | 2,223 | 5,821 |
| Non–current loans and receivables | 3,637 | 5,133 |
| Deferred income tax | 1,847 | 3,608 |
| Non–current deferred charges and accruals | 5,504 | 17,434 |
| Non–current derivative instruments | | |
| Other non–current assets | | |
| Current assets | 89,900 | 357,209 |
| Inventories | 41,646 | 146,093 |
| Current intangible assets | | |
| Biological assets | | |
| Trade receivables | 34,986 | 160,216 |
| Loans and other receivables | | |
| Financial assets | 610 | |
| Cash and cash equivalents | 12,658 | 50,900 |
| Accruals | | |
| Assets from current tax | | |
| Derivative instruments | | |
| Other assets | | |
| Assets held for sale and discontinuing operations | | |
| Called up capital | | |
| Own shares | | |
| **EQUITY & LIABILITIES** | 402,760 | 1,755,951 |
| Equity shareholders of the parent | 288,082 | –944,822 |
| Share capital | 7,500 | 7,500 |
| Called up share capital | | |
| Treasury shares | | |
| Supplementary capital | 204,322 | 244,187 |
| Valuation and exchange differences | 27,229 | 27,145 |

| | | |
|---|---|---|
| Other capitals | | |
| Retained earnings / accumulated losses | 49,031 | –1,223,654 |
| Non–controlling interests | | 26,491 |
| Non–current liabilities | 15,324 | 1,884,337 |
| Non–current liabilities from derivatives | | |
| Non–current loans and borrowings | | |
| Non–current liabilities from bonds | | |
| Non–current liabilities from finance leases | | |
| Non–current trade payables | 2,105 | 98,726 |
| Long–term provision for employee benefits | | |
| Deferred tax liabilities | | 144,178 |
| Non–current provision | | |
| Other non–current liabilities | | |
| Non–current accruals (liability) | 13,219 | 1,641,433 |
| Current liabilities | 99,354 | 789,945 |
| Liabilities from derivatives | | |
| Financial liabilities (loans and borrowings) | | |
| Bond liabilities | | |
| Liabilities from finance leases | | |
| Trade payables | 89,378 | 765,846 |
| Employee benefits | | |
| Current tax liabilities | | |
| Provisions | 9,976 | 24,099 |
| Other liabilities | | |
| Accruals (liability) | | |
| Liabilities related to assets held for sale and discontinued operations | | |
| Date of publication | 2000–03–03 | 2000–03–03 |

### Market value

This data is sourced from two independent, complementing sources. The first is the Warsaw Stock Exchange yearbook, which has been published since 1992. However, prior to 2014, it was only available in PDF format, making it difficult to parse. To overcome this issue, we used a dedicated exporter. Both methods allowed us to export sufficient data to calculate the market value (Eq. 3.1).

$$\text{Market value} = \text{Share price} \times \text{Number of shares} \tag{3.1}$$

The data is presented in millions of PLN currency. The market value for the given year is represented as the value on the last day of the year.

## 3.2 Data Preprocessing

### 3.2.1 Cleaning

During this stage, our goal is to remove invalid and empty data. While some fields may be intentionally left empty depending on the company, others such as the sector must be filled in manually for certain companies. We have also reduced the dataset by removing columns that are consistently empty, such as ’Called up share capital’ or ’Treasury shares’.

### 3.2.2 Transformation to difference

As we are analysing changes in capital structure, we have represented the records as the difference between reports in contiguous years. This allows us to determine how the given balance variable has changed in comparison to the previous year. The market value has also been presented in this way, as described by the equation below (Eq. 3.2).

$$Y_n = \frac{X_n - X_{n-1}}{X_{n-1}} \times 100\% \tag{3.2}$$

where:

- $Y_n$ – Change in year $n$.
- $X_n$ – Value in year $n$.

### 3.2.3 Scaling

The next step is to scale the data. As the data we collected is based on real objects and financial data can be unstable, we have decided to use scaling. We are using a method based on Euclidean distance. Therefore, it is important to avoid situations where one feature has a broad range of values, as this would heavily influence the distance calculation. The simplest method is min–max scaling, which rescales the range of features to [0, 1] or [1, 1] using the equation below (Eq. 3.3).

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \tag{3.3}$$

### 3.2.4 PCA

Principal Component Analysis (PCA) is a widely used algorithm for dimension reduction. It involves projecting data into a lower–dimensional space while preserving information as much as possible. The data is represented by principal components, which are linear combinations of the original variables. The first component represents the largest variance within the dataset. Therefore, the amount of information that we wish to retain can be modified by altering the number of components [37].

### 3.2.5 Encoding

In this section, we encode market values to simplify the problem into binary classification. If the difference in market value between contiguous years is positive, the label will be 1; otherwise, it will be 0.

We also have additional information about the company's sector, which cannot be fully classified as a balance sheet variable. Therefore, we will only check if it improves the prediction metrics. As it is a string variable, we will use One Hot Encoding to present it as multiple columns with binary values instead of one with a string.

## 3.3 Outliers detection

### 3.3.1 Autoencoder

The problem of anomaly detection can be approached using an autoencoder. This model transforms input data into an internal representation and then reconstructs the data on output. If properly trained, it will only reconstruct inlier records. Assuming that anomalies are a minority in the dataset, we can use the Mean Squared Error (MSE) metric as a cost function to separate these two classes. To achieve this, the threshold for classifying a record as an anomaly needs to be defined.

We created a neural network that functions as an autoencoder with input, encoding, decoding, and output layers. The layers consist of dense layers with sigmoid and ReLU activation functions. We implemented this using the TensorFlow framework (Alg. 1).

The data used to train the autoencoder were not processed with PCA, as this neural network can reduce the number of dimensions.

### 3.3.2 ML methods

Numerous machine learning methods are used for predicting anomalies in unsupervised data, this section will outline some of them.

**Algorithm 1:** Autoencoder architecture

```
model = Sequential([
    Input(shape=(63,))
    Dense(40, activation = 'sigmoid')
    Dense(20, activation = 'sigmoid')
    Dense(10, activation = 'relu')
    Dense(20, activation = 'sigmoid')
    Dense(40, activation = 'sigmoid')
    Dense(63)
])
```

### Local outlier factor

The LOF algorithm was first introduced in 2000 [38] to detect anomalies by measuring the local deviation of a given data point from its neighbours. The algorithm introduces the concept of reachability–distance (Eq. 3.4).

$$d_k(A, B) = \max(\text{k-distance}(B), \text{distance}(A, B)) \tag{3.4}$$

The k–distance refers to the maximum distance to the k–nearest neighbours.

The complexity of the method is $n^2$, where n is the number of points and is due to the way we measure LOF for each point (Eq. 3.5).

$$LOF_k(A) = \frac{1}{|N_k(A)|} \sum_{B \in N_k(A)} \frac{LRD_k(B)}{LRD_k(A)} \tag{3.5}$$

Where $N_k(A)$ is the set of k nearest neighbours. While LRD(A) is the local reachability density of an object A defined by (Eq. 3.6). which is the inverse of the average distance of object A from its neighbours.

$$LRD_k(A) = \frac{1}{|N_k(A)|} \sum_{B \in N_k(A)} d_k(A, B) \tag{3.6}$$

By calculating local density, we can identify regions with similar density. This allows us to conclude that points with significantly lower density than their neighbours can be considered outliers. This approach is superior to other methods in terms of detecting local anomalies.

### Isolation Forest

The Isolation Forest concept was proposed in 2008 as an unsupervised prediction algorithm based on decision trees [39]. The algorithm's core function is to create decision trees using random attributes from data. As anomalies are rare and significantly differ from inline points, they will be placed higher in the tree. This results in a shorter path from the root to the anomaly. Achieving each decision tree involves the following steps.

1. Select a sample portion from the original dataset.
2. Choose a random trait from the dataset.
3. Randomly partition the dataset based on the value range of the selected trait.
4. Repeat the last two steps recursively until each observation is isolated.

After creating the specified number of trees, we have trained the Isolation Forest. To identify anomalies, we need to use a metric that is calculated using an equation (Eq. 3.7).

$$S(x,n) = \frac{2 - E(h(x))}{c(n)} \tag{3.7}$$

where:

- $h(x)$ is the number of edges in a tree on the path to $x$.
- $E(h(x))$ is the average value of $h(x)$ from a collection of trees.
- $c(n)$ average value of $h(x)$ with a data set of size $n$.

Having calculated the $S(x,m)$ for each point, we can now assess whether it is an anomaly. After calculating the anomaly score $S(x,m)$ for a given point, we get a value between 0 and 1. As the equation shows (Eq. 3.7), it strongly depends on the length of the path to the point. So if the $S(x,m)$ is close to 1, the path is small, so it can be easily isolated, so we have found an anomaly. If the value is less than 0.5, the path is large, which represents an outlier data point. By analysing the whole dataset, we can also say that the sample has no anomalies if the score is around 0.5.

#### One–class SVM

The Support Vector Machine (SVM) was first published in 1995 [40]. It is a machine learning algorithm that maps data into a multidimensional feature space, enabling the categorization of points even when they are not linearly separable. The algorithm finds a separating boundary between classes and then transforms the data so that a separating hyperplane can be drawn. In the mathematical context, SVMs use kernel methods to transform data. The choice of kernel influences how the algorithm operates. Linear, polynomial, or RBF kernels are the most popular choices. An SVM with a one–class allows for the separation of normal data from anomalies. One of the greatest strengths of SVM is its ability to handle large datasets.

### 3.3.3 Metrics

As we are predicting anomalies without labels, we will rely on visual methods. We can easily plot multidimensional data using PCA or t–SNE.

T–Distributed Stochastic Neighbour Embedding is a statistical method that maps high–dimensional data points by assigning them a location in a two or three–dimensional map. The algorithm first creates a probability distribution over pairs of multidimensional objects, assigning higher probability to similar objects. The algorithm minimises the divergence between the probability distributions of the original high–dimensional and lower–dimensional spaces using gradient descent. This optimises the lower–dimensional embedding to a stable state, resulting in clusters of points that were correlated in the multidimensional space [41].

## 3.4 Market value prediction

Once the preliminary processing phase has been completed, the subsequent step is to predict market value through the application of various ML methodologies.

### 3.4.1 Join dataset

Up to this point, we have been transforming two datasets independently. Our next step is to merge the data with the labels. The capital–property structure dataset includes a variable called 'End of period', while the market value dataset includes measurement data. We will match these datasets based on their respective dates.

### 3.4.2 Convolutional NN

This is a regularised type of feed–forward neural network that typically comprises three layers. The first layer is the convolutional layer, which performs a dot product between two matrices. One of these matrices is called the kernel and has learnable parameters, while the other is the bounded part of the receptive field. It recognises spatial hierarchies and local patterns in the data. The pooling layer replaces the network's output at certain locations by deriving a summary statistic of nearby outputs, thereby reducing the spatial size of the representation. The pattern recognition in the image is invariant to translation, meaning that it can be recognized regardless of its position. The fully connected layer links each neuron in one layer to every neuron in the next layer. This layer combines all the high–level features to the ultimate forecast.

CNNs are commonly utilised for image processing. Therefore, we will convert our numerical data into images. The following steps outline this process. The NN architecture is illustrated in the figure below (Fig. 3.3). When a regression task is considered, the activation function in the final layer is modified from a sigmoid function to a linear function.

### Correlation matrix creation

Given a data set with n columns, we calculate the correlation matrix and receive a matrix of size n x n.

### Optimisation of the pixel position in the correlation matrix

Currently, the variables in the correlation matrix are arranged as columns in the input data set. The objective is to rearrange them so that more correlated fields are closer together. This can be achieved by using Simulated Annealing, a probabilistic technique for approximating the global optimum of a given function. In this case, the energy function is defined as follows (Eq. 3.8).

$$E = \sum_{(i,j)\in P} |c[R(i), R(j)]| \cdot d(i,j), \tag{3.8}$$

where the distance is

$$d(i,j) = \{x(i) - x(j)\}^2 + \{y(i) - y(j)\}^2, \tag{3.9}$$

where $i$ and $j$ are the indices for pixels, and $x(i)$, $y(i)$ represent the $x$ and $y$ coordinates of pixel $i$, respectively. $P$ represents the set of all combinations of pixels. $R(i)$ represents the financial ratio corresponding to pixel $i$, and $c[R_1, R_2]$ represents the correlation coefficient between financial ratios $R_1$ and $R_2$.

At each step of the method, two pairs of variables are sampled from the correlation matrix. The energy function is then calculated for both the original and switched positions. The new arrangement is accepted based on the acceptance function (Alg. 2).

The process is repeated until the temperature exceeds the defined limit. During each iteration, the temperature value is multiplied by a factor within the interval of (0,1).

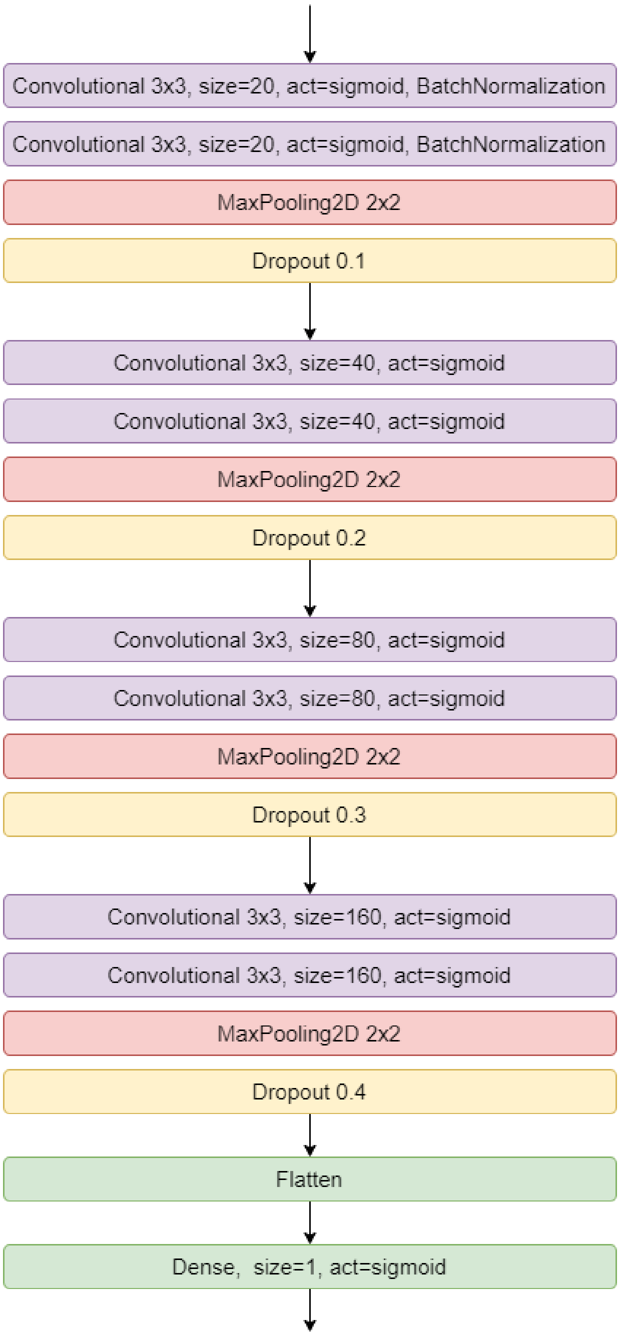
Convolutional 3x3, size=20, act=sigmoid, BatchNormalization
Convolutional 3x3, size=20, act=sigmoid, BatchNormalization
MaxPooling2D 2x2
Dropout 0.1
Convolutional 3x3, size=40, act=sigmoid
Convolutional 3x3, size=40, act=sigmoid
MaxPooling2D 2x2
Dropout 0.2
Convolutional 3x3, size=80, act=sigmoid
Convolutional 3x3, size=80, act=sigmoid
MaxPooling2D 2x2
Dropout 0.3
Convolutional 3x3, size=160, act=sigmoid
Convolutional 3x3, size=160, act=sigmoid
MaxPooling2D 2x2
Dropout 0.4
Flatten
Dense, size=1, act=sigmoid

Figure 3.3: Convolutional Neural Network architecture [42]

**Algorithm 2:** Simulated Annealing Acceptance Function

```
Input  : Change in energy $\Delta E$, current temperature $T$, cooling schedule parameter $t$
Output: Acceptance probability $P$
Initialize $k_B$ as Boltzmann constant;
if $\Delta E < 0$ then
    $P \leftarrow 1$;
end
else
    $P \leftarrow e^{-\Delta E/(k_B \cdot T \cdot t)}$;
end
```

#### Pictures creation

With the correlation matrix optimised, we can now create a dictionary that maps each pair of variables to its corresponding position in the picture. As we are considering pairs of parameters, we will transform the current data into ratios. Therefore, when the map dictionary for variables $a$ and $b$ returns position $(x, y)$, the value of pixel $(x, y)$ in the created image will be $a/b$. These values are then mapped to a range of [0, 255] to represent brightness.

#### Training process

With the generated picture, we can now commence training the neural network to predict market value. The results will be compared with other approaches. This approach is a modification of the method proposed in a publication that focuses on bankruptcy prediction [32].

### 3.4.3 Dense neural network

Given the complexity of the problem at hand, we have opted to employ a dense neural network as our approach. The network's architecture consists of blocks, each of which is constructed from two layers.

1. The dense layer is defined as the layer where every neuron is connected to every neuron in the previous layer. This trait renders them not only universal, but also computationally extensive to train. They are capable of functioning properly with preprocessed big datasets, identifying global patterns. However, they do not address data sequencing. As an activation function in this layer, we employed ReLU and sigmoid.

2. A dropout layer is incorporated into each block in order to prevent the neural network from overfitting. Overfitting occurs when the model learns to memorise the details

of the training dataset rather than to fit the global patterns. The incorporation of a dropout layer mitigates overfitting by randomly dropping a fraction of neurons during training. This is of particular importance in the context of the limited amount of data available. The fraction of neurons that are dropped is controlled by hyperparameters.

Once the layers have been defined, it is necessary to select the activation function. This is a function that is applied to every neuron output. The objective is to reduce linearity within the data, which allows the creation of more nested models for training. The most commonly used activation functions are sigmoid and rectified linear units (ReLU). In our model, ReLU was selected as the activation function in every layer, apart from the final layer, where we used sigmoid. The diagram below illustrates the usage of the activation function (Fig. 3.4). The Dense Neural Network is also employed in regression tasks. In this instance, the activation function in the final layer is not sigmoidal, but linear. The architecture was implemented with Tensorflow.

## 3.4.4 ML methods

### Grid search

Grid search is a method that optimises the performance of an ML model by identifying the most effective combination of hyperparameters. It involves testing a range of potential configurations and selecting the most suitable one. Despite increasing the complexity of the process, this approach can lead to enhanced learning outcomes. All models described in this section were learned with usage of this method.

### Logistic Regression

This statistical model, also known as the logit model, employs a linear transformation to combine linear independent variables with the logit function. This process returns a linear value from the (0,1) interval, which represents odds. Odds are defined as the probability of success divided by the probability of failure, as illustrated by the following equations (Eq. 3.10), (Eq. 3.11).

$$\mathrm{Logit}(\pi) = \frac{1}{1 + \exp(-\pi)} \tag{3.10}$$

$$\ln\left(\frac{\pi}{1-\pi}\right) = \beta_0 + \beta_1 X_1 + \ldots + \beta_k X_k \tag{3.11}$$

The beta parameter in this model is commonly estimated via maximum likelihood estimation (MLE). The objective is to identify a set of weights that maximises the probability of observing the actual classification results based on the data [43].

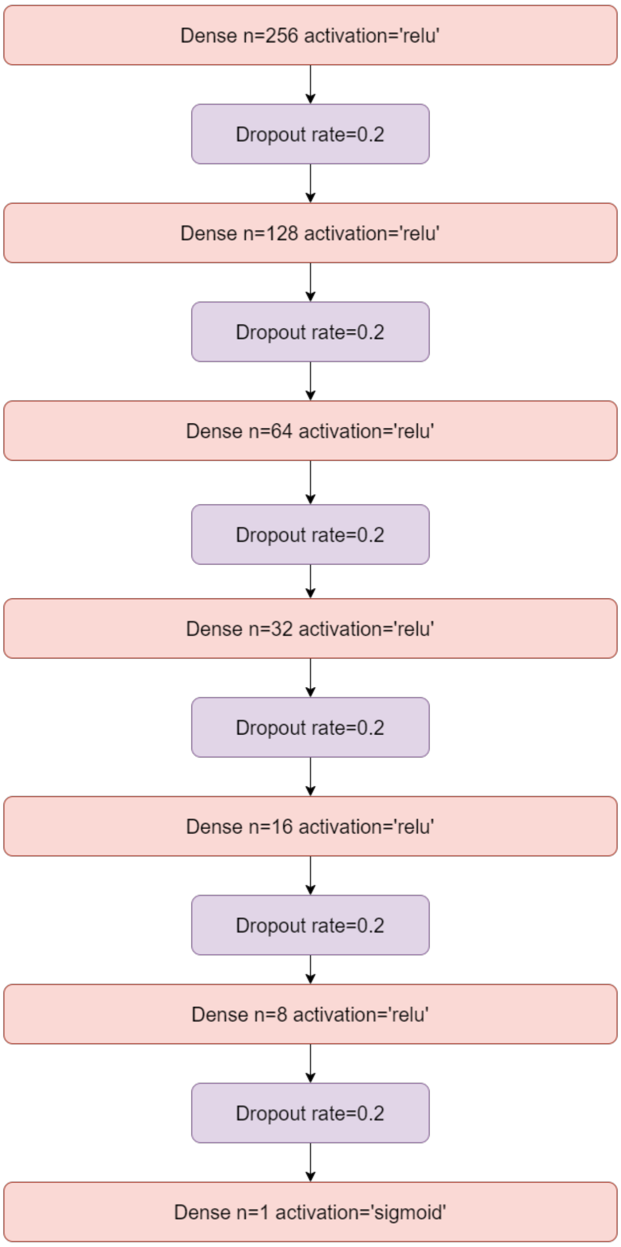
Dense n=256 activation='relu'
Dropout rate=0.2
Dense n=128 activation='relu'
Dropout rate=0.2
Dense n=64 activation='relu'
Dropout rate=0.2
Dense n=32 activation='relu'
Dropout rate=0.2
Dense n=16 activation='relu'
Dropout rate=0.2
Dense n=8 activation='relu'
Dropout rate=0.2
Dense n=1 activation='sigmoid'

Figure 3.4: Dense neural network architecture

### SVM

The SVM will be employed not only for the anomaly detection section, where it was fully described in (Sec. 3.3.2), but also to predict market value.

### Decision Tree

Decision tree (DT) is a non–parametric supervised learning method used for classification and regression. A DT comprises a root node, internal nodes and leaf nodes. The internal node represents a division based on the available features. The lead nodes represent all the possible outcomes within the dataset. This method is based on the divide and conquer strategy, whereby the optimal split within a tree is identified. The creation of a tree is represented by the following steps:

- The split feature selection is based on a given metric, which allows us to identify the optimal split within the dataset. This can be achieved by utilising the Gini index (Eq. 3.12) or entropy (Eq. 3.13) metrics.
- The dataset is divided based on the selected feature dataset. This process is repeated recursively until one of the stop conditions is met. Some of these conditions are maximum tree depth or minimum number of samples in leaf.

$$G = 1 - \sum_{i=1}^{C} p_i^2 \tag{3.12}$$

$$E = -\sum_{i=1}^{C} p_i \log_2 p_i \tag{3.13}$$

where $p_i$ is the probability of an instance belonging to class $i$.

The construction of this tree allows us to predict samples by traversing it in a top–down approach.

### K–Nearest Neighbours

The k–Nearest Neighbours (KNN) algorithm is a non–parametric, supervised learning classifier. It is a lazy learner, which means that it does not learn a discriminative function from the training data, but instead memorises the training dataset. It was first developed in 1951 by Fix and Hodges [44] and it is based on the following steps:

1. The selection of the hyperparameter k.
2. The distance between each neighbour is calculated and the k closest are identified.
3. It is recommended that the sample's class be specified as the most frequently occurring class in the neighbours set.

### 3.4.5 Metrics

As our sole objective is classification as well as regression, we will use the metrics below to compare the selected methods. These metrics are defined by equations,

where:

$TP$ – True positives – prediction, and real value are true.

$TN$ – True negatives – prediction is true while real value is false.

$FP$ – False positive.

$FN$ – False negative.

$y_i$ – Actual value of observation i.

$\hat{y}_i$ – Predicted value of observation i.

$n$ – Number of observations.

#### Accuracy

$$\frac{TP + TN}{TP + TN + FP + FN} = \frac{correct\ predictions}{all\ predictions} \tag{3.14}$$

#### Precision

$$\frac{TP}{TP + FP} \tag{3.15}$$

#### Recall

$$\frac{TP}{TP + FN} \tag{3.16}$$

#### F1 Score

$$\frac{2 * precision * recall}{precision + recall} \tag{3.17}$$

#### ROC Curve

The ROC curve is a plot in the unit square that connects points with coordinates determined by recall and precision for various cut–off levels. The AUC, the area under the ROC curve, is a popular measure of classifier discriminative power.

### Confusion matrix

The confusion matrix is a tool used to evaluate the quality of binary classification, which involves classifying data into two categories. The data is labelled as either positive or negative and is then classified into either a predicted positive or predicted negative class.

### MAE

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{3.18}$$

### MAPE

$$\mathrm{MAPE} = \frac{100\%}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{3.19}$$

### MSE

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{3.20}$$

### MSLE

$$\mathrm{MSLE} = \frac{1}{n}\sum_{i=1}^{n}(\log(1 + y_i) - \log(1 + \hat{y}_i))^2 \tag{3.21}$$

# 4 Experimental Results

This part will present the findings of the research described in the previous chapter (Chap. 3). The following section will present the steps that will be encountered in the experiments.

1. The initial phase of the analysis involved the preprocessing of the data and the creation of visual representations of the results.

2. The efficacy of anomaly detection algorithms in our case, as demonstrated by the PCA and T-SNE methods, will be evaluated through visualisation.

3. The regression task will be approached with previously defined ML methods. The aim is to compare and select the best prediction algorithm and dataset.

4. The optimal dataset and model will be employed to facilitate a comparative analysis of anomaly detection models.

5. Should the regression methods yield unsatisfactory results, classification methods will be used. The previously selected anomaly detection methods will be utilised.

## 4.1 Data preprocessing

### 4.1.1 Datasets used in experiments

Given that we are utilising multiple datasets, it is necessary to import them all and concatenate them correctly. The data originates from various sources, and therefore a portion of it is of no use because it does not comply with other sources. In order to merge the capital—property structure with the market value, it is necessary to match not only the year but also the company name. In order to create the final record, it is necessary to have valid records from two siblings' years for the same company. It should be noted that not all the aforementioned conditions can be met for all records. The table below (Tab. 4.1) illustrates the number of records per preprocessing step. The final number of records prior to the deletion of anomalous data is 1963.

Table 4.1: Number of records in selected preprocessing stage

| **Preprocessing stage** | **Number of records** |
|---|---|
| Companies | 349 |
| Capital–property structure | 3471 |
| Market value | 7800 |
| Capital–property structure with market value | 2188 |
| **Difference between sibling records** | **1963** |

### 4.1.2 Dataset visualisation

The dataset was preprocessed using t–SNE and PCA methods to facilitate visualisation. However, the inclusion of additional information, such as year and company name, which are not strictly part of the capital–property structure, resulted in distortion of the data visualisations. This was due to the fact that year and company name exhibited the greatest variation among all the other variables. Consequently, PCA information reduction to 97% was achieved by leaving only five variables (Fig. 4.1), (Fig. 4.2).

Having established the veracity of this claim, we proceeded to remove them from the training dataset, basing our analysis exclusively on capital–property structure. In this instance, the data clusters exhibited a reduced degree of distinction (Fig. 4.3).

### 4.1.3 PCA

In our data set, comprising 63 traits, we were able to reduce the number of dimensions to 46 while maintaining 97% of the original information (Fig. 4.4).

### 4.1.4 Encoding

In order to train our classification models, market value was encoded. Upon analysing the preliminary prediction results, it was determined that values should be divided into two categories. This was due to the fact that the accuracy level was not satisfactory for the two labels, therefore there was no need to divide them further. Consequently, every market value was assigned as either a positive or negative value category. The comparison of the number of records within these classes is presented below (Fig. 4.5). The ratio between the two classes is well–balanced, with the classes being nearly evenly split.

## 4.2 Anomaly detection

The t–SNE visualisation method was employed to assess the efficacy of the algorithms. This approach is the only viable means of evaluating the effectiveness of the algorithms,

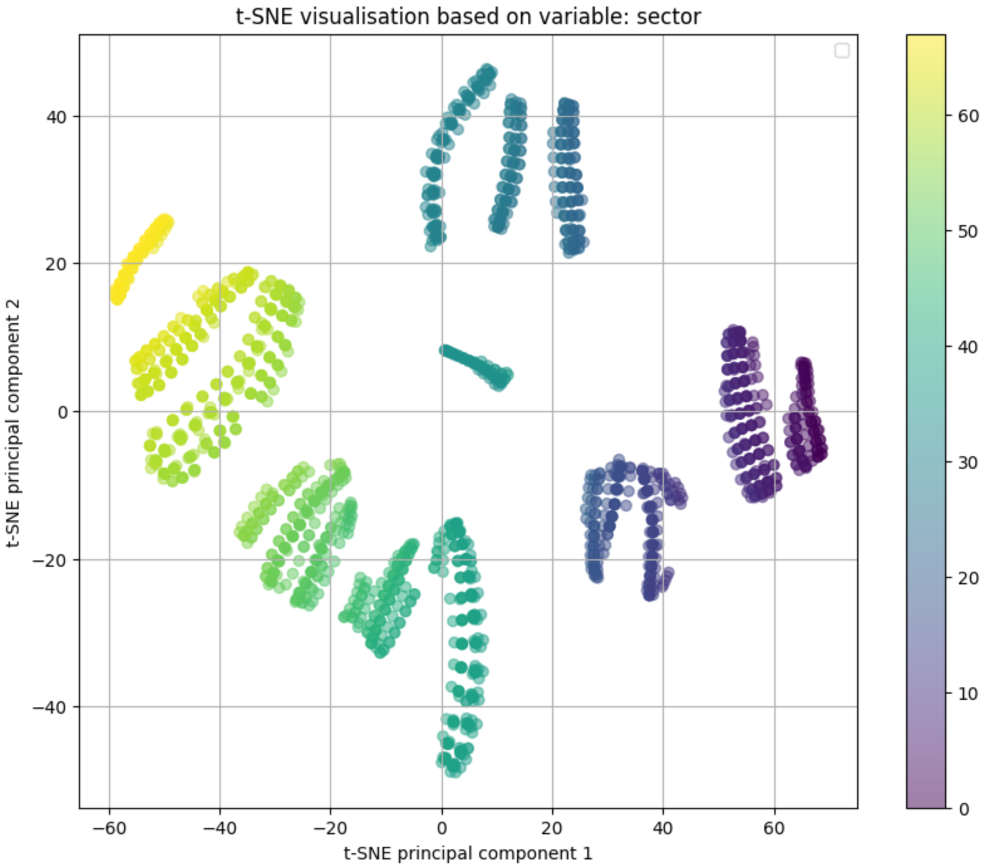

Figure 4.1: Data with year and sector variables visualized with t–SNE. Colours represent sector

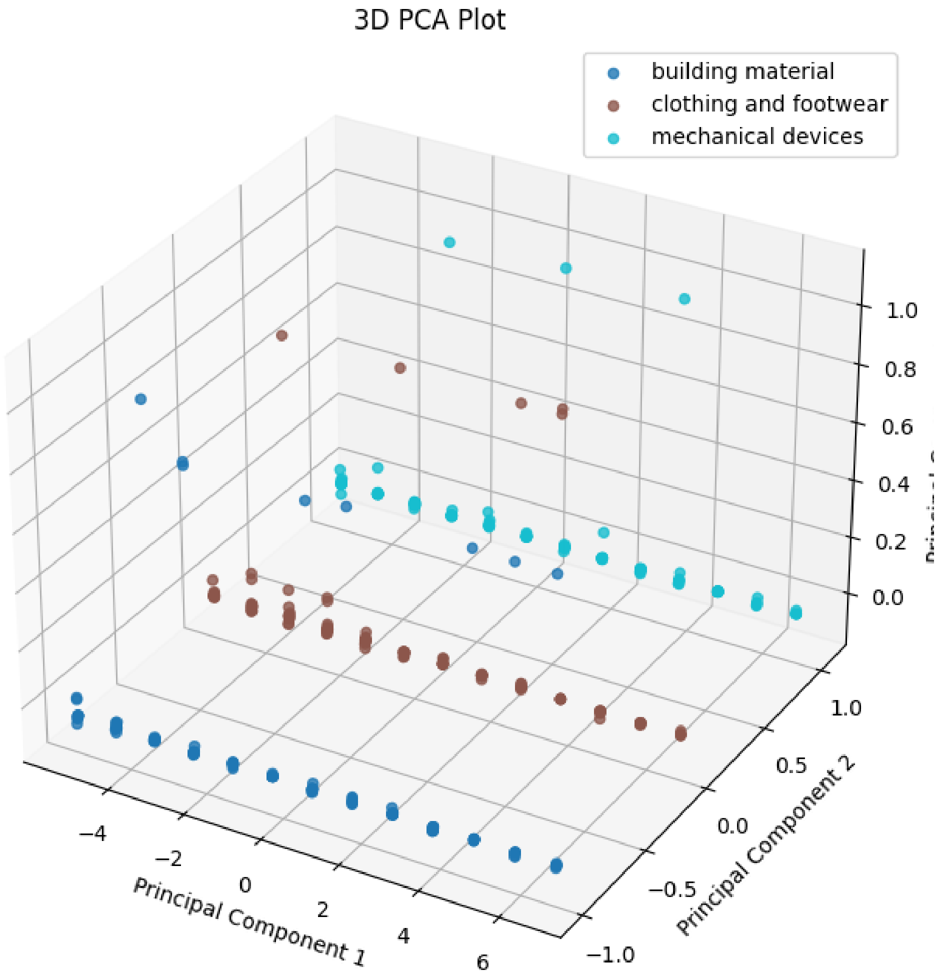

Figure 4.2: Data with year and sector variables visualized with 3–dimensional PCA for most popular sectors

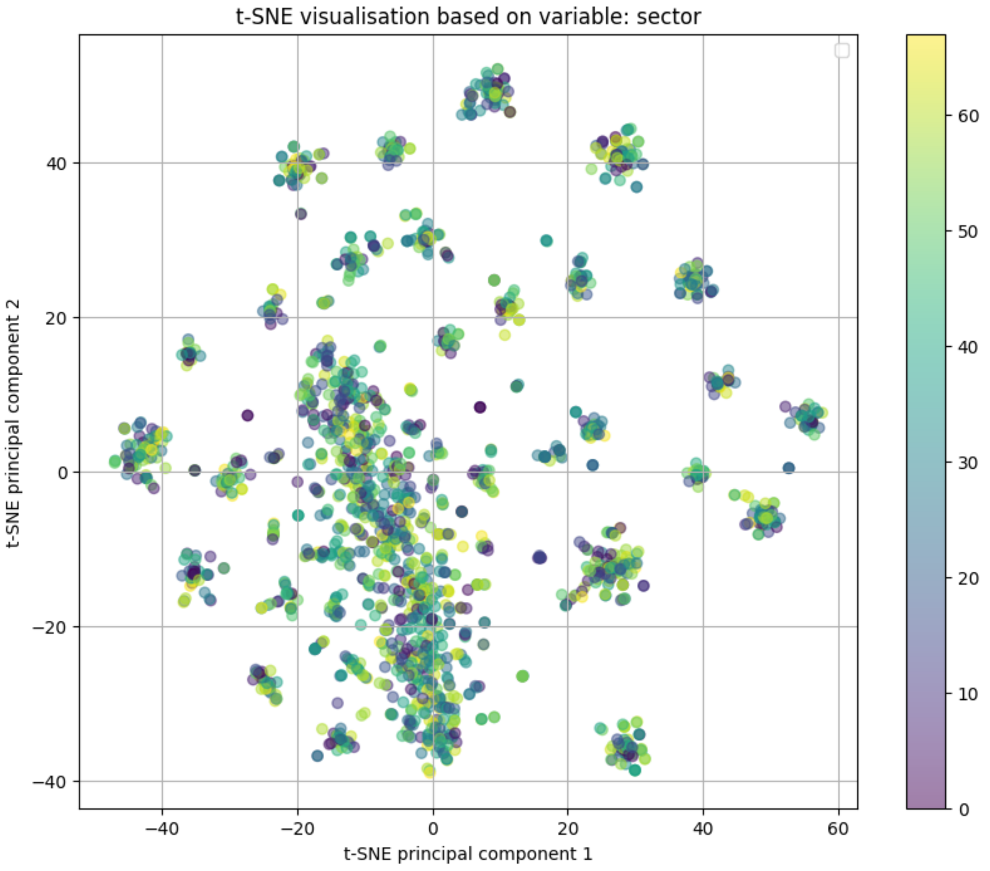

Figure 4.3: Final data visualized with t–SNE. Colours represent sectors

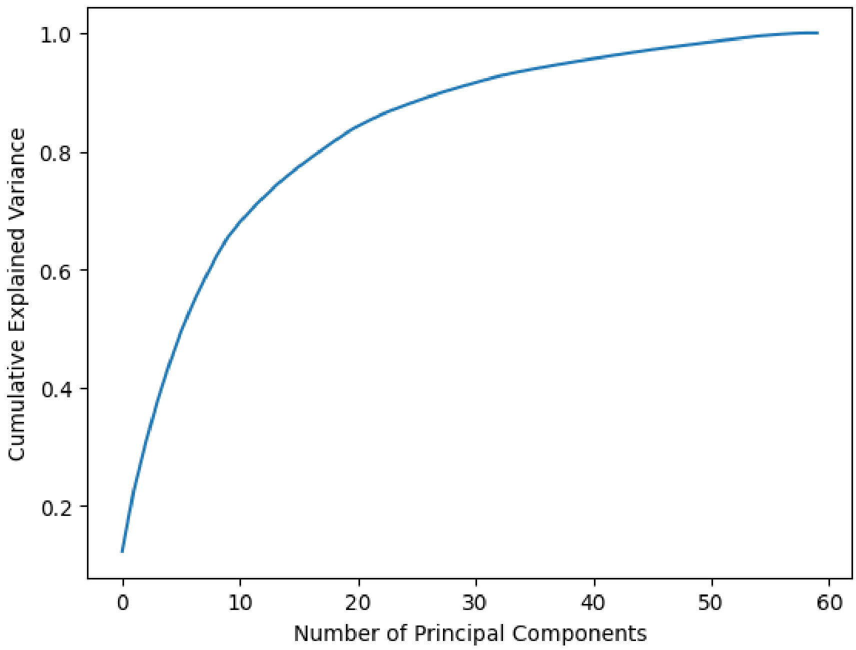

Figure 4.4: The proportion of information that depends on the number of principal components

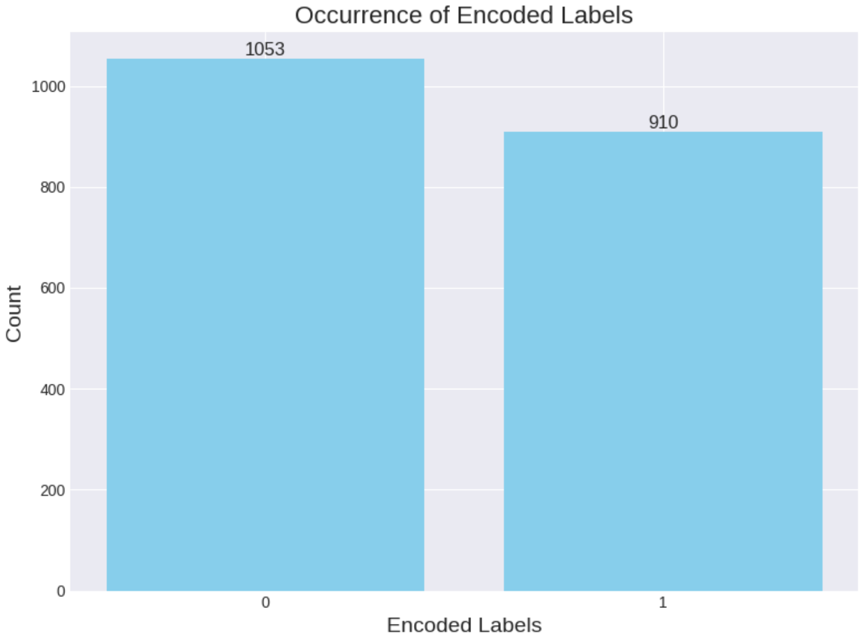

Figure 4.5: The number of instances of encoded labels per class.

given that the data is unlabelled. The results demonstrate minimal differences between the algorithms, although these differences are likely to be attributable to the distinct approaches employed by each algorithm (Fig. 4.6). The specific differences between the algorithms will be described in detail for each method in the following subsections. It is important to note that the visualised models are trained on data that has undergone a t–SNE transformation. This may result in slight variations in the observed results.

### 4.2.1 Isolation Forest

The Isolation Forest anomaly detection strategy is based on the distance from the tree root to the record leaf in the tree. This distance was therefore visualised. On the chart, outliers that were classified due to a high root–leaf distance are presented. The chart shows that the Isolation Forest is capable of finding global anomalies (Fig. 4.7).

### 4.2.2 Local Outlier Factor

The objective of this algorithm is to identify local outliers, which are evident in the presented example. This algorithm produces the most disparate results compared to other algorithms. In this example (Fig. 4.6), where global outliers and local outliers are clearly distinguishable, the discrepancy is strikingly apparent. In the detailed chart (Fig. 4.8), the LOF metric is presented for each record. Records with the highest metric value are then classified as outliers. The presented chart comprises companies from one of the most popular sectors.

### 4.2.3 One–Class SVM

The One–Class SVM with RBF kernel is dividing the space. Using t–SNE, this multidimensional division can be presented on a two–dimensional plane (Fig. 4.9). It can be observed that records outside the selected space are assigned as outliers. Multiple divisions allow the identification of local anomalies. This can be modified by modifying the gamma hyperparameter in order to distinguish only one subspace. In this case, records outside will be global outliers.

### 4.2.4 Autoencoder

The objective of the autoencoder is to identify the optimal training data. The optimiser employed is Adam, and the loss function is MSE. The learning curve, presented in the figure below (Fig. 4.10), indicates that MSE is decreasing, which signifies that the model is developing in an appropriate manner. To ensure the model's efficacy, it was assumed that

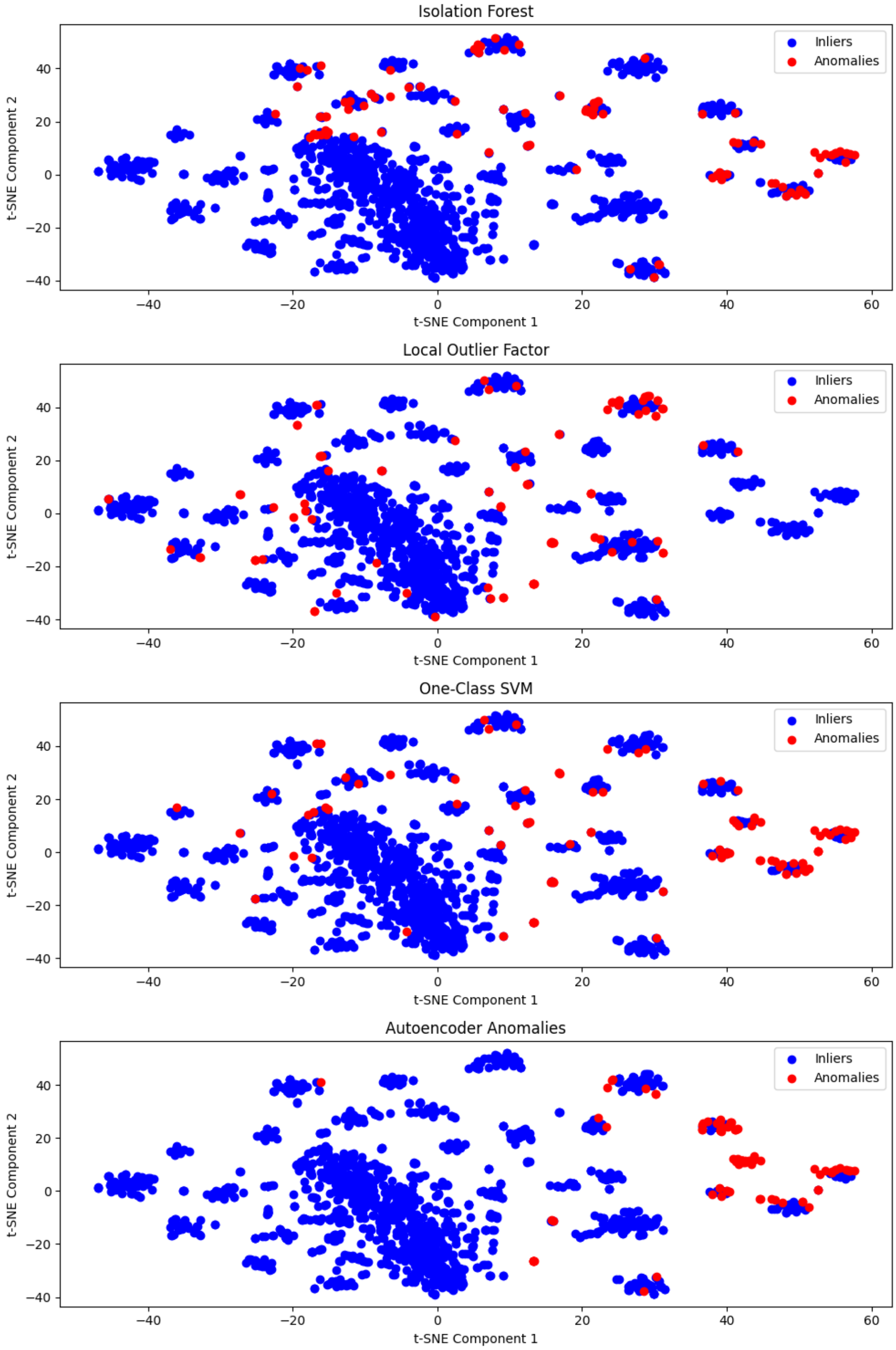

Figure 4.6: A comparative analysis of anomaly detection visualisation techniques utilising various algorithms

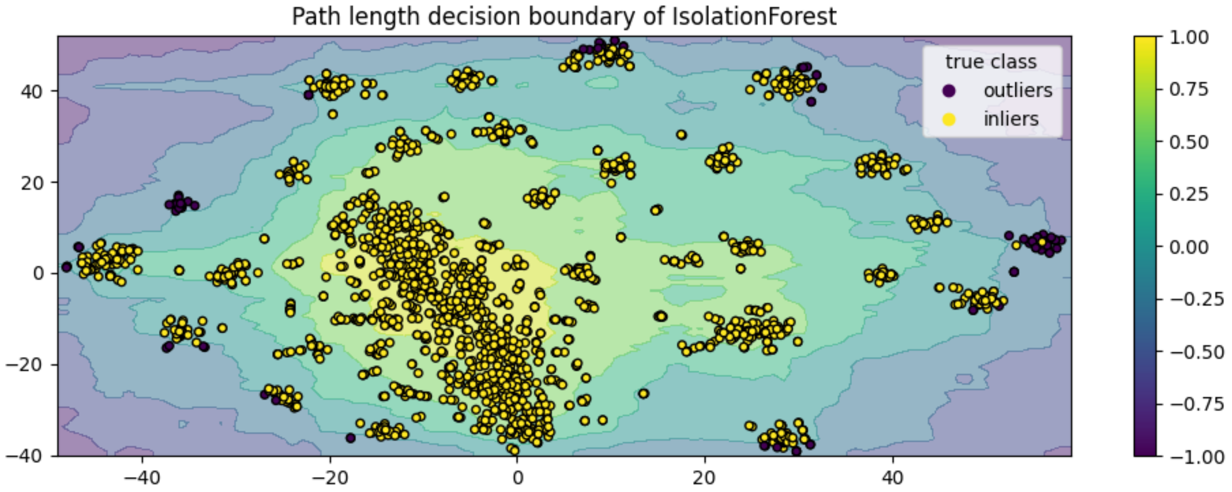

Figure 4.7: Isolation Forest algorithm division visualisation

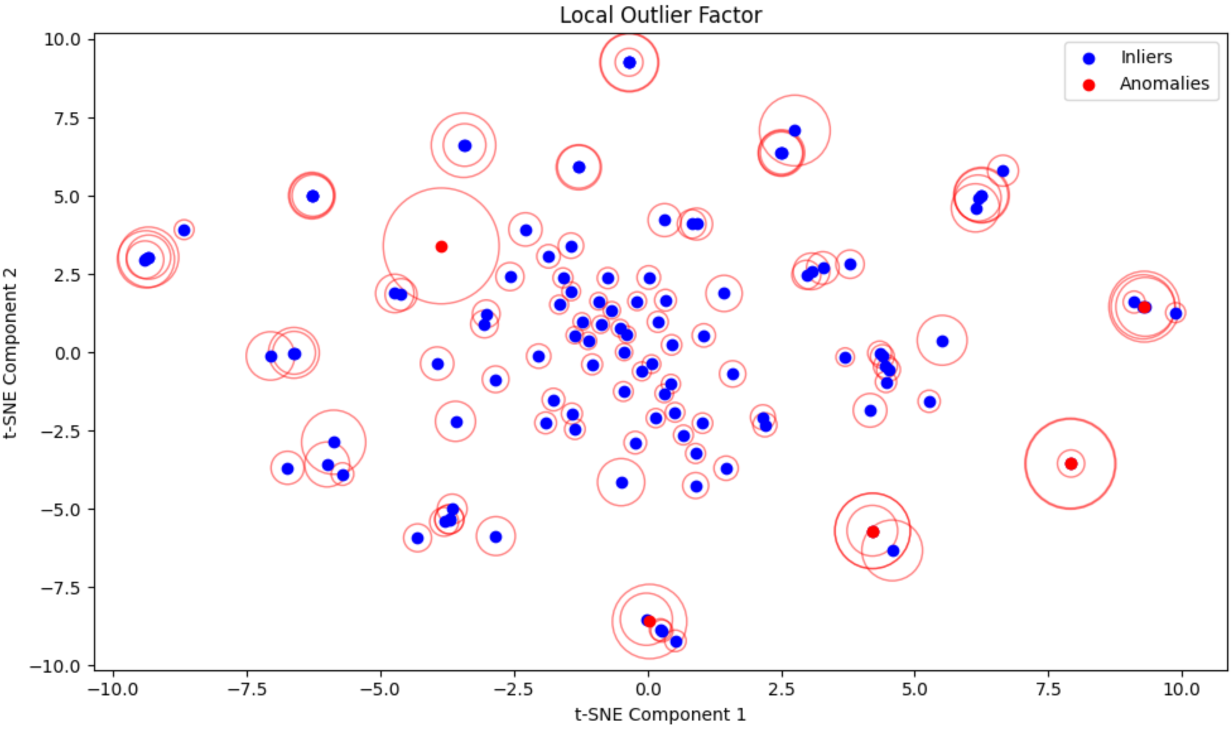

Figure 4.8: Local Outlier Factor algorithm division visualization

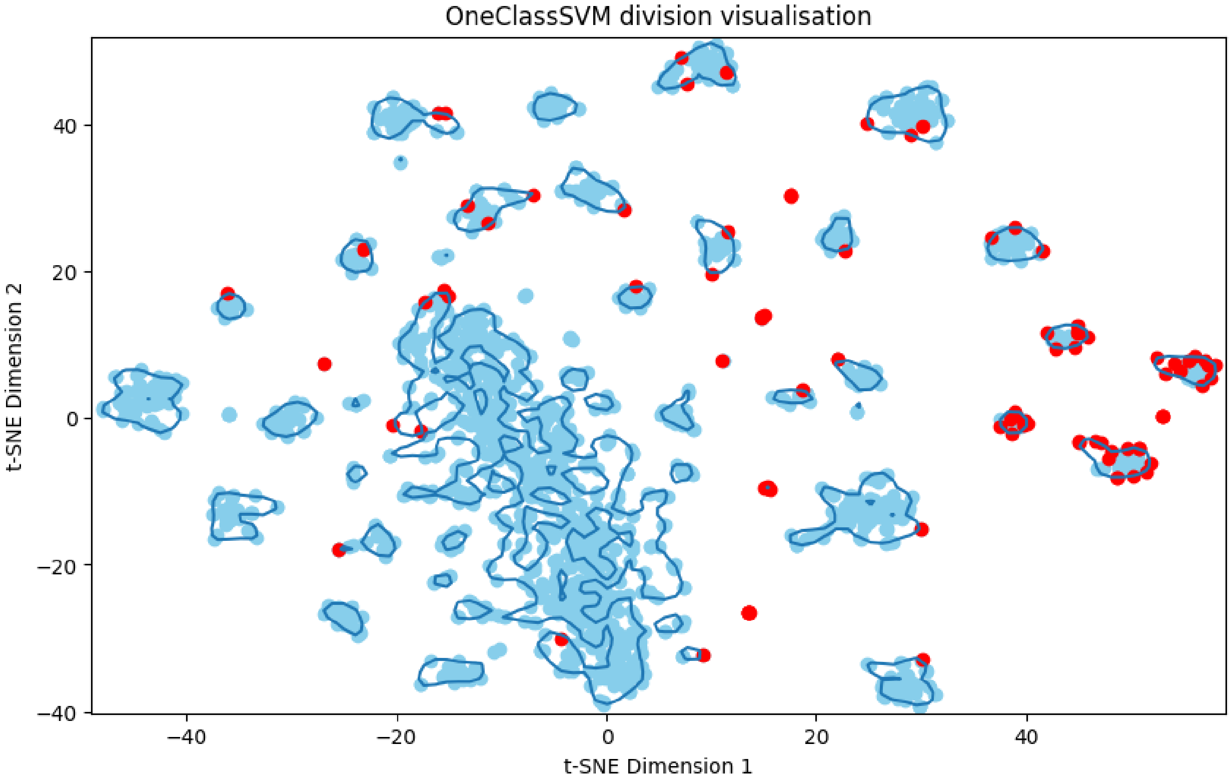

Figure 4.9: One–Class SVM algorithm division visualization

5% of the data were outliers. The model primarily identified global anomalies within the dataset (Fig. 4.6).

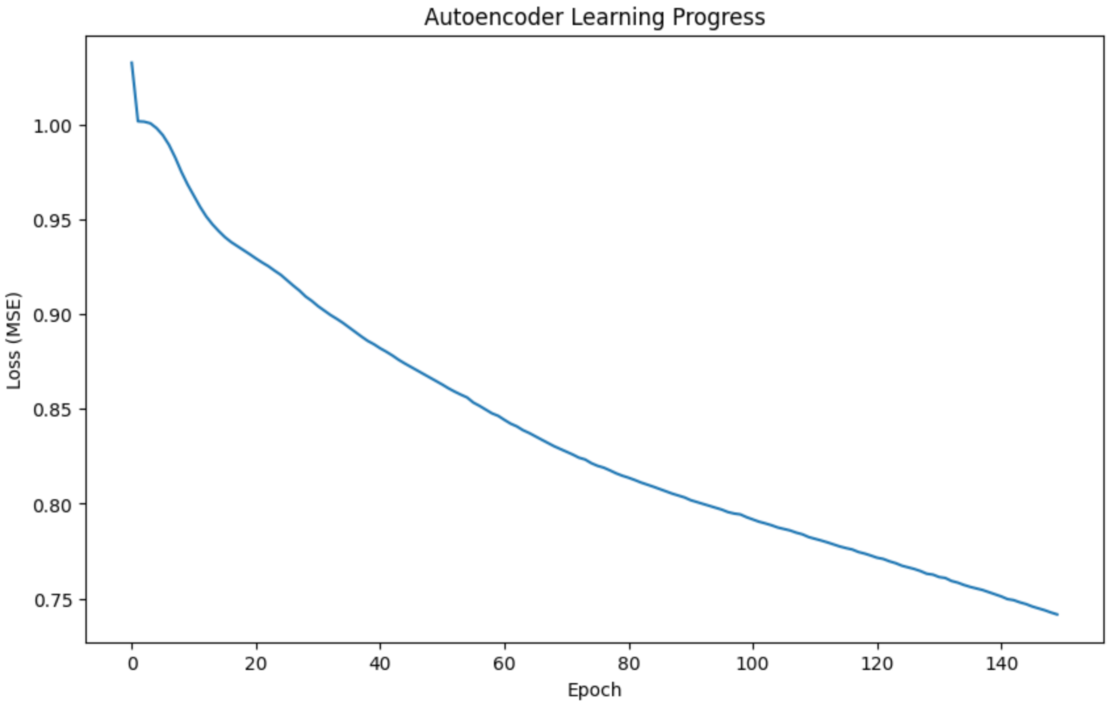

Figure 4.10: Autoencoder learning curve

## 4.3 Market value prediction

This section presents the results of the previously mentioned models, which have undergone classification and regression testing.

### 4.3.1 Training data

A limited number of datasets have been generated through the aforementioned transformations and will be employed in forthcoming experiments.

1. **Transformed** – The records are presented as the difference between sibling records, as described in (Sec. 3.2.2)

2. **After PCA** – “Transformed” dataset after PCA transition.

3. **Correlation images** – This dataset represents a modification of the original "Transformed" dataset, which was not subjected to principal component analysis (PCA) and was subsequently purged of anomalies. Such records were then converted to images. Each variable is positioned on the image according to a pixel mapping, which was created using simulated annealing based on correlation. This approach results in more correlated data being positioned closer together. Consequently, the image displays discernible shapes (Fig. 4.12) that were generated in lieu of randomly positioned pixels (Fig. 4.11).

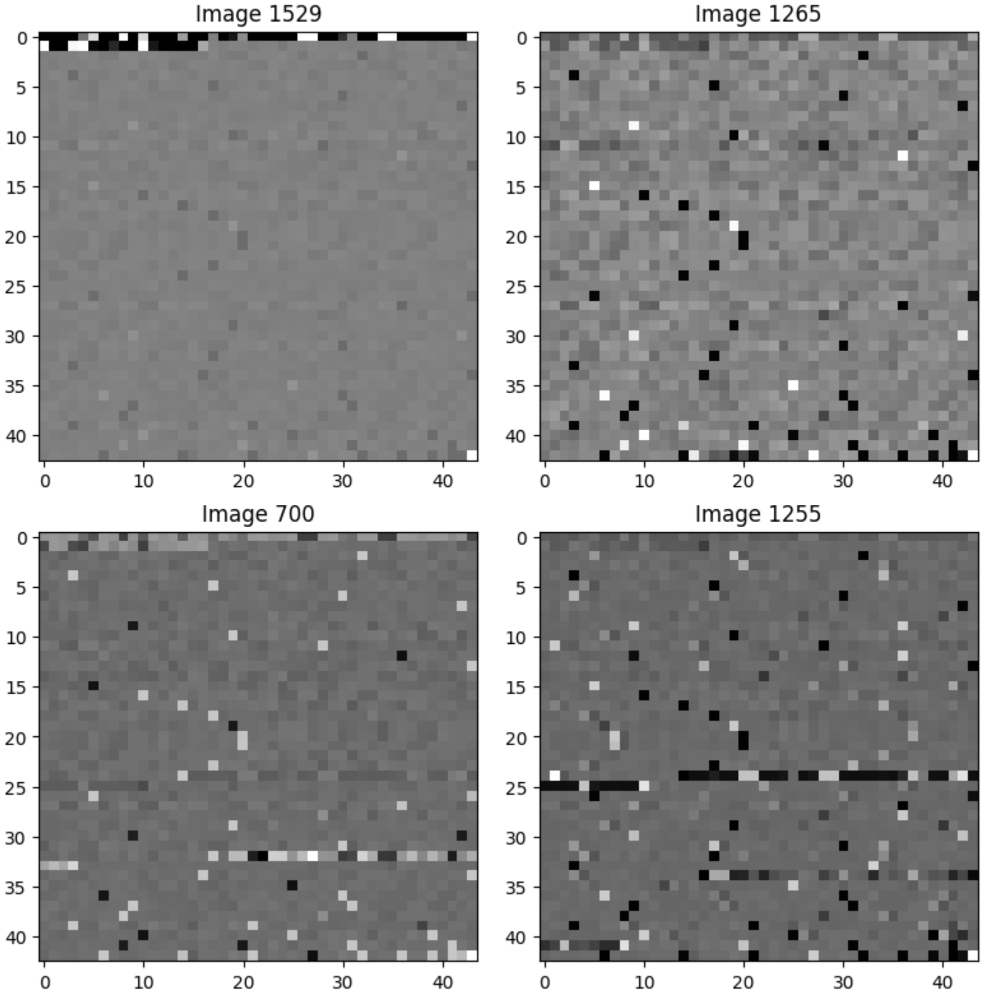

Figure 4.11: Visualisation of random images before simulated annealing

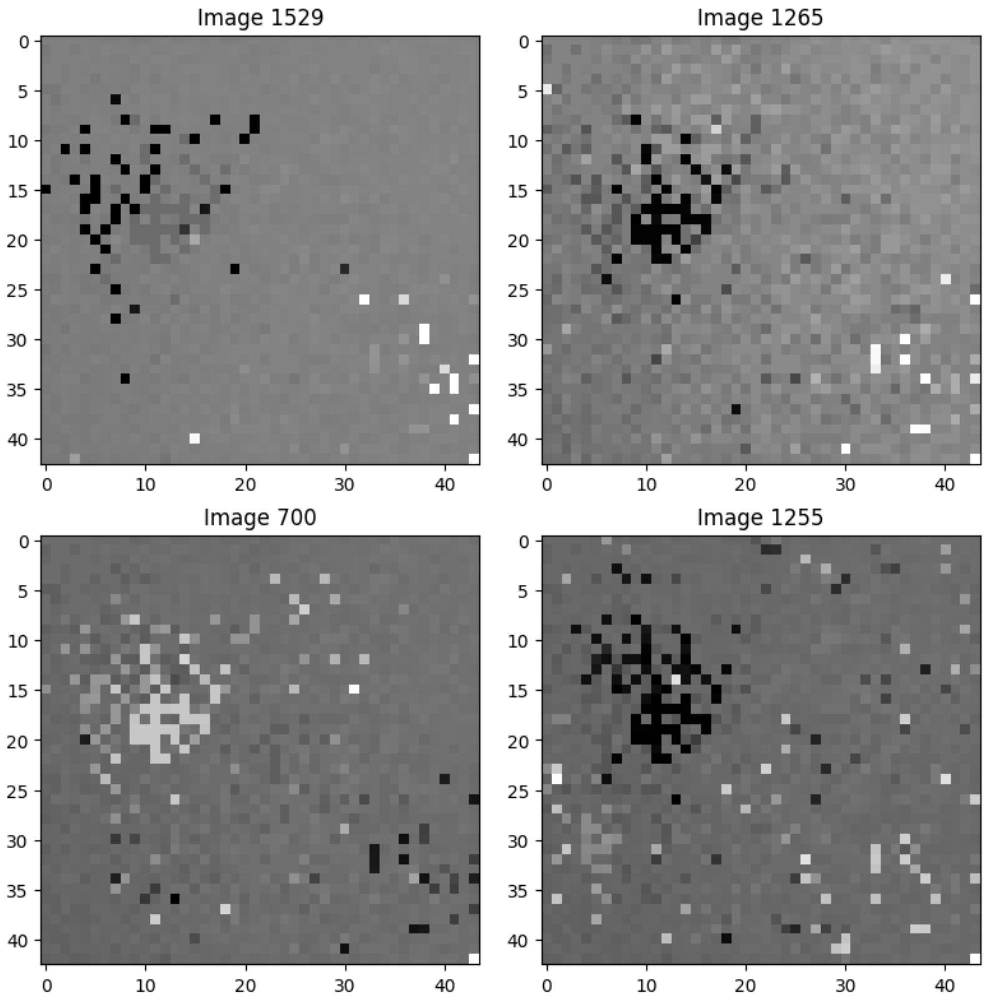

Figure 4.12: Visualisation of random images after simulated annealing

### 4.3.2 Regression

The objective of this section is to evaluate the efficacy of various regression models. The aim is to identify the optimal model and dataset for our specific case. Additionally, the results will inform our decision regarding the necessity of employing classification models contingent on regression approach performance. The predicted value is the company market value, expressed as a percentage increase relative to the previous year.

Having defined the metrics and the model, we present the results of the test in the following tables, for each metric: MEA (Tab. 4.2), MSE (Tab. 4.3), MSLE (Tab. 4.4), MAPE (Tab. 4.5). Metrics are used to compare not only different models, but also datasets. Additionally, example learning curves were presented to illustrate the learning process. These curves were generated for a Deep Neural Network with a "Transformed" dataset (Fig. 4.13) and a "After PCA" dataset (Fig. 4.14). Furthermore, a Convolutional Neural Network (CNN) was employed in this task. However, due to the high complexity of the simulated annealing (SA) algorithm employed to map data to images, we were only able to test the architecture on a single dataset. The learning process was visualised with this model and the 'Correlation images' dataset, as shown in the figure below (Fig. 4.15).

Table 4.2: MEA metric for various models and datasets

| MAE | | | |
|---|---|---|---|
| **Dataset** | **Linear Regression** | **Deep NN** | **Convolutional NN** |
| Transformed | 75.22 | 48.63 | – |
| After PCA | 52.72 | 41.54 | – |
| Correlation images | – | – | 40.33 |

Table 4.3: MSE metric for various models and datasets

| MSE | | | |
|---|---|---|---|
| **Dataset** | **Linear Regression** | **Deep NN** | **Convolutional NN** |
| Transformed | 58195.50 | 13138.47 | – |
| After PCA | 19317.92 | 4597.77 | – |
| Correlation images | – | – | 4390.80 |

The results demonstrate that the PCA method enhances the performance of the considered models. This is particularly evident in the MSE metric, where the value of the metric was reduced from 13138.47 to 4597.77. Another consistent trend in the results is that Deep NN outperforms Linear Regression. This is anticipated given that Deep NN is considerably more complex than Linear Regression and is therefore better suited to our needs. The discrepancy is particularly evident in the MAPE (Fig. 4.5) metric, where the error is approximately two

Table 4.4: MSLE metric for various models and datasets

| MSLE | | | |
|---|---|---|---|
| **Dataset** | **Linear Regression** | **Deep NN** | **Convolutional NN** |
| Transformed | 5.36 | 5.27 | – |
| After PCA | 5.49 | 3.94 | – |
| Correlation images | – | – | 4.95 |

Table 4.5: MAPE metric for various models and datasets

| MAPE | | | |
|---|---|---|---|
| **Dataset** | **Linear Regression** | **Deep NN** | **Convolutional NN** |
| Transformed | 392.93 | 249.43 | – |
| After PCA | 276.12 | 155.07 | – |
| Correlation images | – | – | 261.15 |

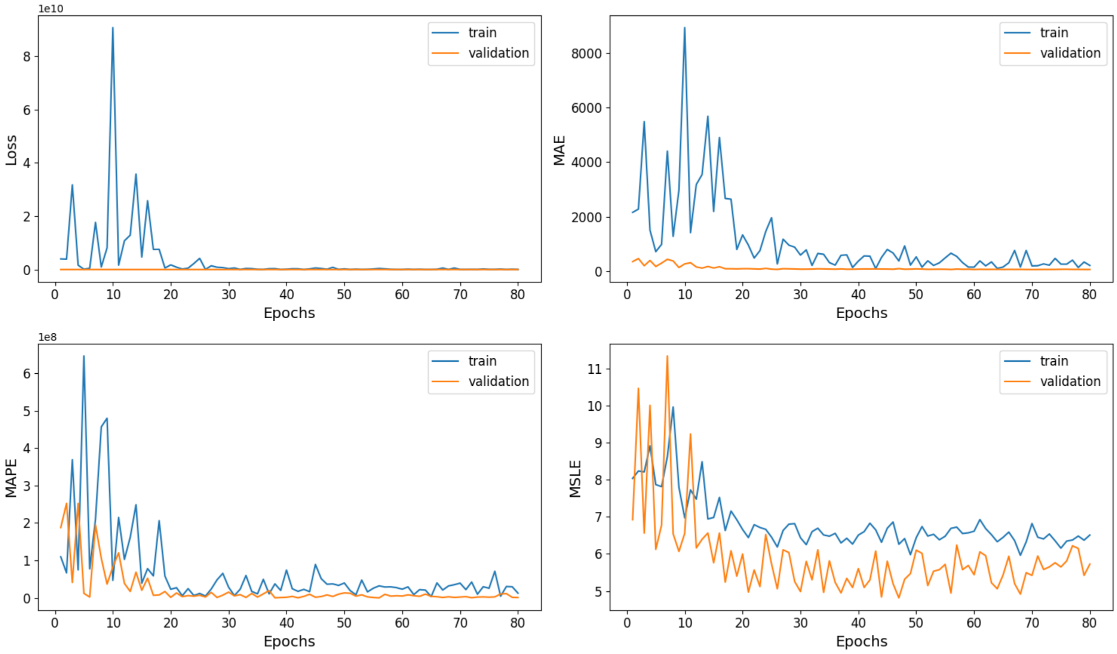

Figure 4.13: Deep Neural Network learning curve for "Transformed" dataset

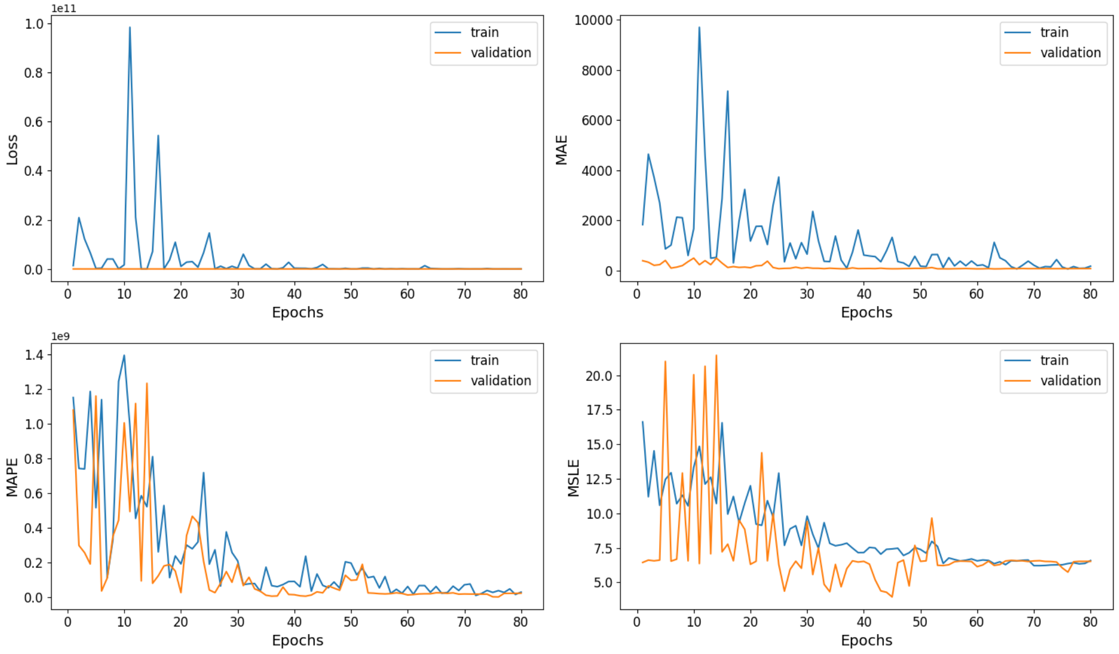

Figure 4.14: Deep Neural Network learning curve for "After PCA" dataset

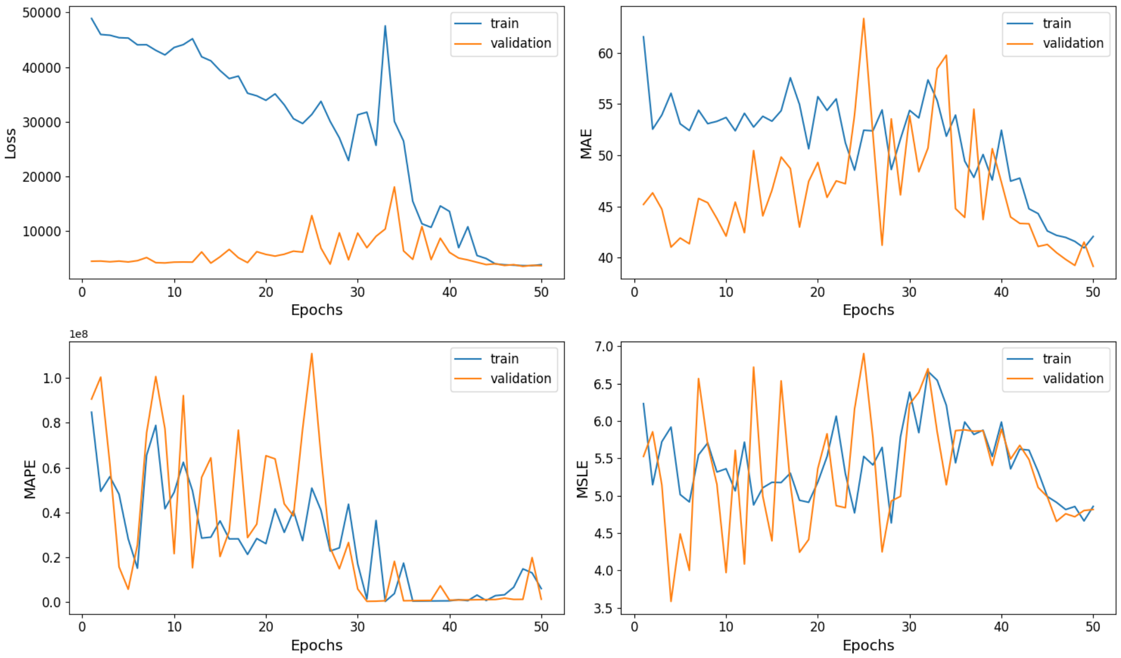

Figure 4.15: Convolutional Neural Network learning curve for "Correlation images" dataset

times greater. When comparing the performance of the Convolutional NN and the Deep NN, it is not possible to determine which model performed better. In terms of MAE (Fig. 4.2) and MAE metric (Fig. 4.2), the deep neural network (NN) demonstrated superior performance, while the convolutional NN exhibited greater accuracy in MAPE (Fig. 4.5) and MSLE metric (Fig. 4.4). Furthermore, the observed differences were relatively minor.

A review of the learning curves for NN models reveals that the results have undergone a transformation. It can be observed that the training and validation learning curves (Fig. 4.13) are monotonically decreasing, and both are converging. These are the defining characteristics of a stable machine learning model. The learning curves of convolutional neural networks (CNNs) (Fig. 4.15) exhibit less stability than those of deep neural networks (DNNs), as illustrated in (Fig. 4.13). It is evident that the validation set exhibits considerable fluctuations, which may be attributed to the inadequacy of the dataset or the lack of sufficient preparation. Nevertheless, it can be observed that the loss function is monotonically decreasing for both the training and validation curves.

The results of the prediction are unsatisfactory, with a high value of errors. Therefore, it is not possible to perform a regression task with such a dataset and configuration. The MAE metric is approximately 50% in this experiment, indicating that on average, in every prediction, the result is defined as real value +/- 50%. Consequently, when the value of the company increased by 50% in a given year, the model can predict values in the range of 0% to 100%. Such a result is unacceptable, therefore we will attempt to delete anomalies and change the task to classification.

### 4.3.3 Anomaly detection methods comparison with prediction models

Anomaly detection models were compared with t-SNE method visualisation. However, the model and dataset selected in the previous subsection will be employed to compare the results with selected metrics. The Deep Neural Network model and the ”After PCA” dataset will be used to run the prediction with and without the anomaly deletion. The results of such an experiment are presented below (Tab. 4.6). Additionally, the selected learning curves for the aforementioned experiment are presented (Fig. 4.16).

The results indicate that anomaly deletion significantly improves prediction results. In most cases, the error rate has decreased. The only exception is the OCSVM method, which, according to MAE and MSLE metrics, slightly worsens the model results. The autoencoder model outperforms the others, as evidenced by its enhanced learning process, as illustrated in (Fig. 4.16). The learning curve is significantly more smooth after anomaly deletion than before (Fig. 4.14).

In the preceding section, we demonstrated the efficacy of anomaly detection algorithms through visual means. In this section, we employed metrics to corroborate the results. The autoencoder was identified as the optimal choice for our purposes and will be utilized in the subsequent experiments.

Table 4.6: Anomaly detection methods comparison

| **Method** | **With anomalies** | **Without anomalies** |
|---|---|---|
| | **MAE** | |
| Local Outlier Factor | 41.54 | 40.47 |
| Isolation Forest | 41.54 | 39.75 |
| One-Class SVM | 41.54 | 44.61 |
| Autoencoder | 41.54 | 37.89 |
| | **MSE** | |
| Local Outlier Factor | 4597.77 | 4469.25 |
| Isolation Forest | 4597.77 | 4294.98 |
| One-Class SVM | 4597.77 | 4934.62 |
| Autoencoder | 4597.77 | 4024.95 |
| | **MSLE** | |
| Local Outlier Factor | 3.94 | 3.91 |
| Isolation Forest | 3.94 | 3.88 |
| One-Class SVM | 3.94 | 3.83 |
| Autoencoder | 3.94 | 3.71 |
| | **MAPE** | |
| Local Outlier Factor | 155.07 | 153.77 |
| Isolation Forest | 155.07 | 141.03 |
| One-Class SVM | 155.07 | 138.37 |
| Autoencoder | 155.07 | 131.77 |

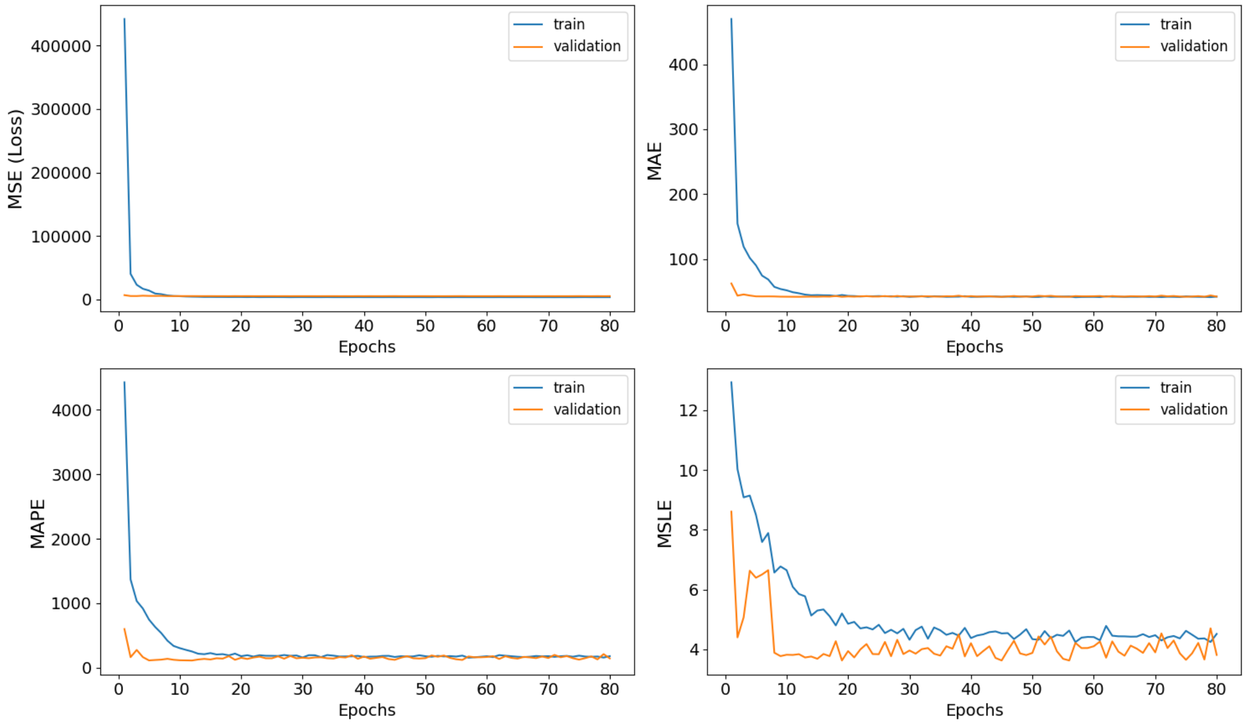

Figure 4.16: Deep NN learning curve after anomaly deletion with autoencoder

### 4.3.4 Classification

As previously discussed, the regression results were deemed unsatisfactory. Consequently, the problem was transformed into a classification problem, with the objective of determining whether the market value will increase or decrease. A selection of machine learning (ML) methods was employed to analyse a dataset following a principal component analysis (PCA) transition and the removal of anomalous data. The performance of the ML models was evaluated using classification metrics, including accuracy, F1 score, precision, and recall. Furthermore, it is possible to compare the models with a random classifier, examining a chart that presents the ROC AUC (Fig. 4.17).

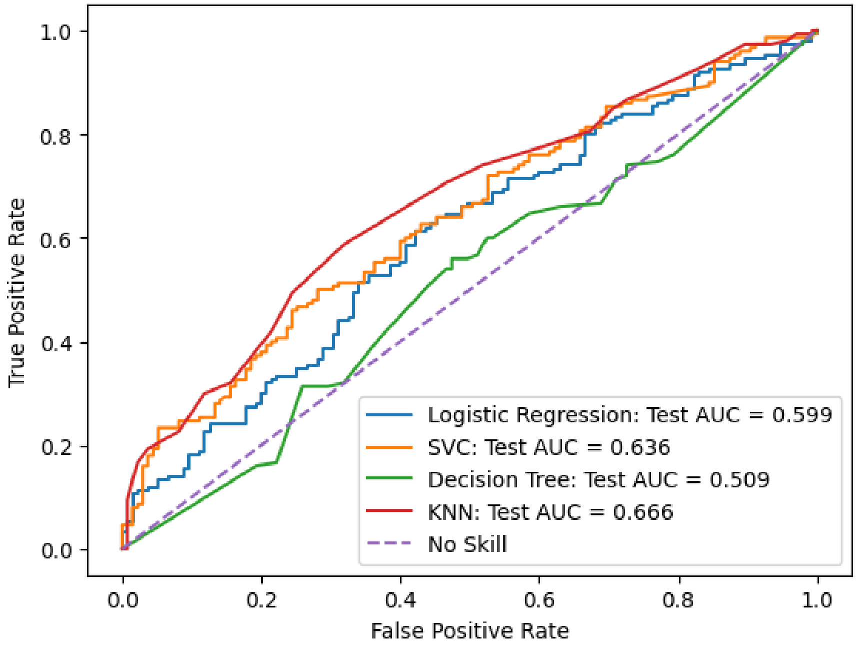

Figure 4.17: ROC AUC for ML models

In addition to the use of conventional machine learning (ML) methods, such as decision trees and random forests, this study employed more sophisticated deep neural networks (DNNs) and convolutional neural networks (CNNs). The various ML approaches are delineated in the following sections, and their performance is compared in the table below. (Tab. 4.7)

#### Logistic Regression

The model achieved an accuracy of 59%, which is not the optimal result. However, it could be argued that it is superior to a random classifier. The detailed classification results are

Table 4.7: Classification results

| **Model** | **Accuracy** | **F1** | **Precision** | **Recall** |
|---|---|---|---|---|
| Logistic Regression | 0.592 | 0.623 | 0.608 | 0.640 |
| SVC | 0.582 | 0.502 | 0.674 | 0.4 |
| Decision Tree | 0.523 | 0.544 | 0.547 | 0.540 |
| KNN | 0.618 | 0.671 | 0.613 | 0.740 |
| Deep NN | 0.608 | 0.573 | 0.594 | 0.583 |
| Convolutional NN | 0.639 | 0.610 | 0.591 | 0.631 |

presented in a confusion matrix (Fig. 4.18).

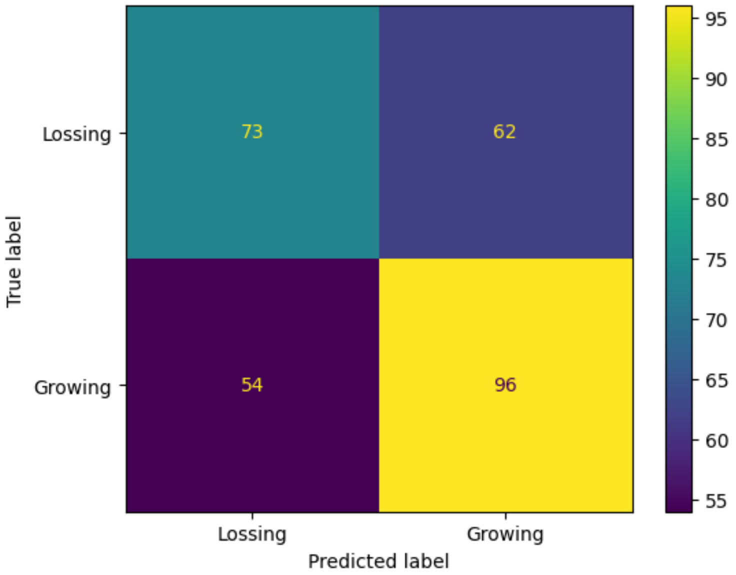

Figure 4.18: Logistic Regression confusion matrix

### SVC

The model demonstrated an accuracy rate of 58%. The confusion matrix indicates that the majority of errors were due to the misclassification of the ’Growing’ class (Fig. 4.19).

### Decision Tree

The model achieved an accuracy of only 52%, which is comparable to that of a random classifier. This indicates that the model was unable to effectively address the problem, re-

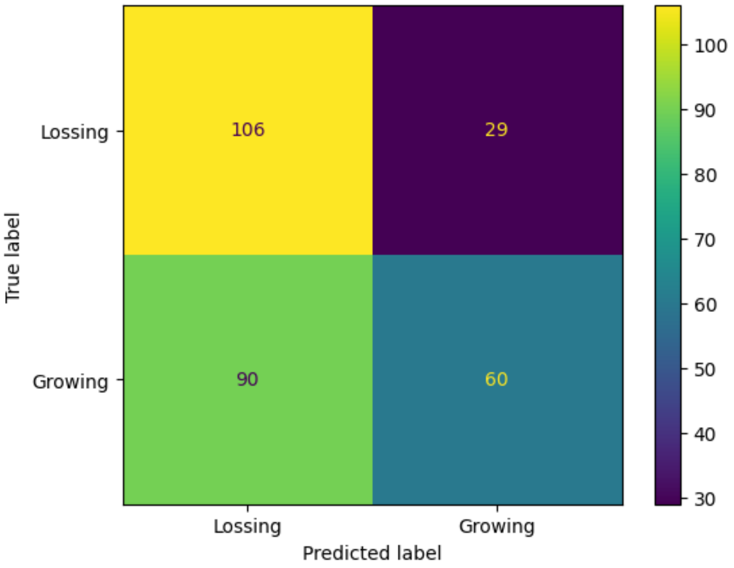

Figure 4.19: SVC confusion matrix

sulting in unsatisfactory outcomes. The confusion matrix demonstrates that the data was distributed evenly across all classes (Fig. 4.20).

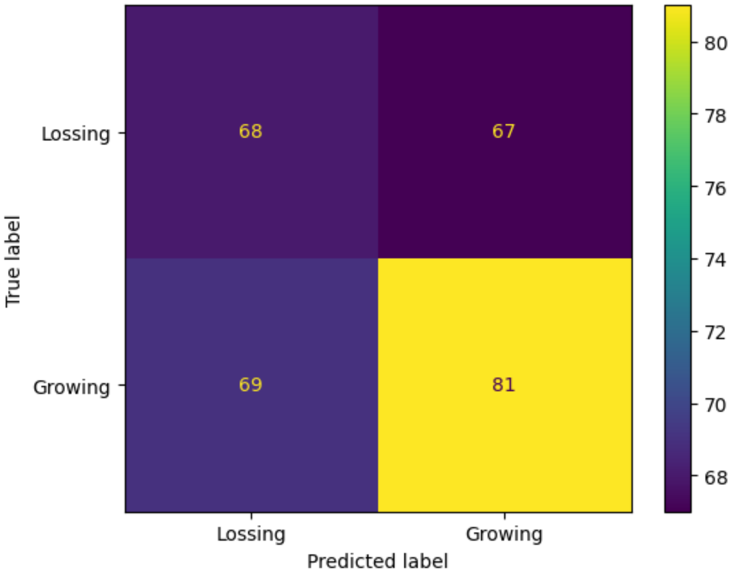

Figure 4.20: Decision Tree confusion matrix

### K–Nearest Neighbors

The model achieved an accuracy of 62%, which represents the highest score among all models. The confusion matrix indicates that a significant number of records were incorrectly classified as 'Growing' (Fig. 4.21).

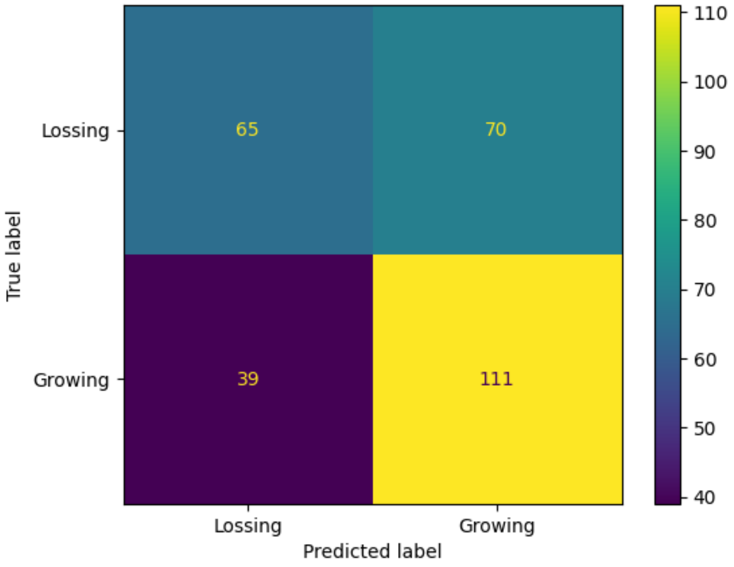

Figure 4.21: K–Nearest Neighbors confusion matrix

### Deep Neural Network

The model achieved an accuracy of 60%. The distribution of incorrect predictions across classes is approximately equal, as illustrated by the confusion matrix. (Fig. 4.22).

### Convolutional Neural Network

Once the images have been prepared, they are used to train a convolutional neural network to classify the data sets. The learning process was observed with a graph showing its main metrics (Fig. 4.23). The model achieved 64% accuracy. In the further development, it was decided to limit the number of epochs by looking at the validation loss graph, which shows the model's overfitting.

The results were also presented in the form of a confusion matrix, as shown below (Fig. 4.24).

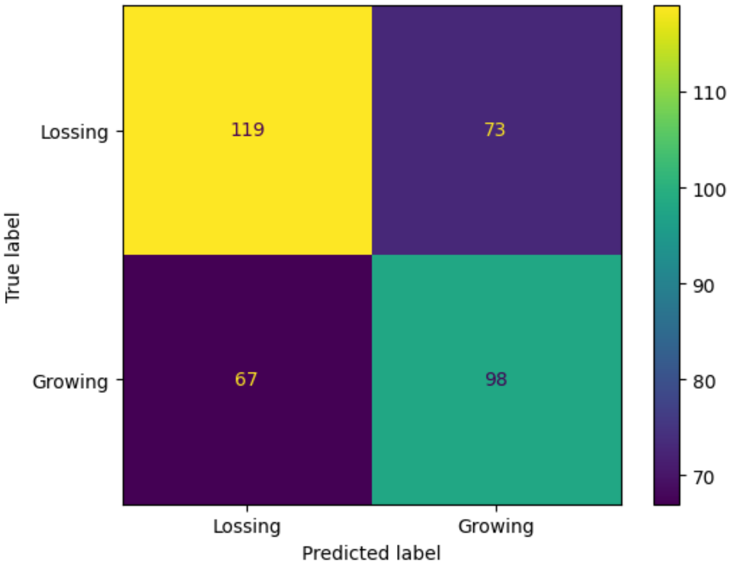

Figure 4.22: Deep Neural Network confusion matrix

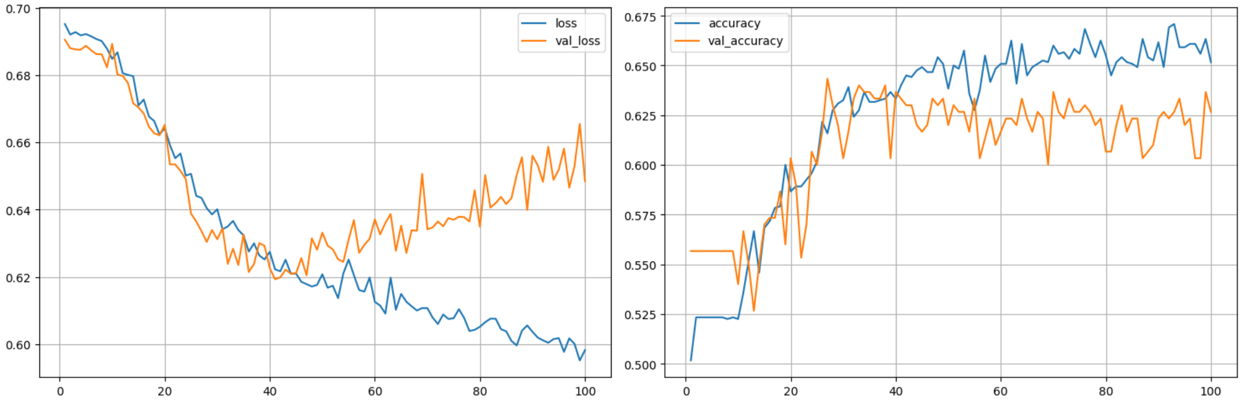

Figure 4.23: Convolutional Neural Network learning process

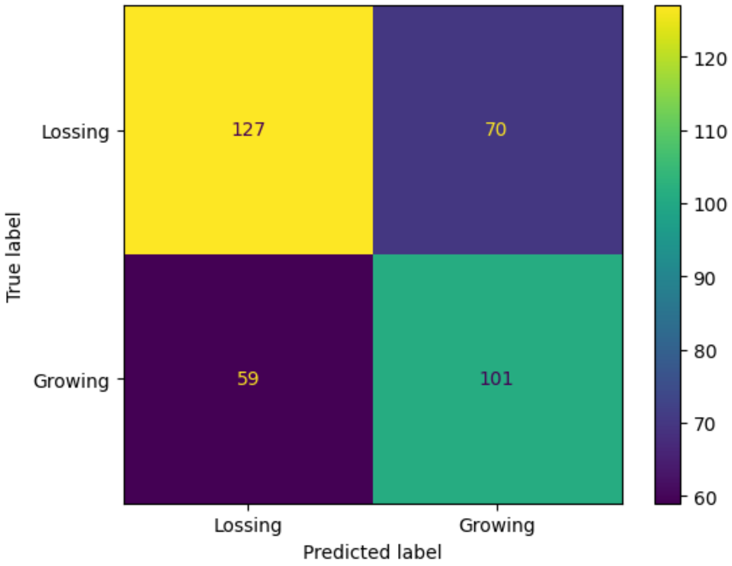

Figure 4.24: Convolutional Neural Network confusion matrix

# 5 Conclusions

This chapter is the conclusion of the results we achieved in the previous chapter, as well as a summary of the issue raised in the thesis. We will also outline the next steps possible to encounter to extend the works.

Before the work could begin, proper research was needed. Once we had identified the economic aspects, we were able to transform our data. The correctness of this process was visualised using PCA and t–SNE methods, which are crucial when working with unlabelled data. With this method implemented, we could approach the core problem.

The first part of the work, aimed at detecting and removing anomalies before further data processing, could be considered successful based on the results. Different methods with different problem–solving approaches have been used. They aim to detect whether local or global anomalies. All were visualised and concluded with results. It was also proven that removing the anomalies improved the forecast results. Therefore, taking into account the nature of our data, we can assume that an appropriate anomaly detection method has been selected and applied.

The second part of the thesis, which includes market value prediction, does not provide sufficient arguments to confirm the thesis. Seeing bad performace of regression task we decided to transform it to classification. All the models used were similar in their results, all of them achieving around 60% accuracy, which is too low to prove a strong correlation. Although we have used a wide variety of methods, from typical ML methods to complex convolutional neural networks, all the models have similar results. This gives us confidence in the validity of each of them.

Nevertheless, research in this area should continue. It is difficult to say why the hypothesis was not confirmed. It could be whether a problem with the data, or the correlation does not exist. However, the results show that the approach is correct, so it can be applied to other datasets with such data.

As the topic is very complex, there are many ways to explore it further. Here are some of them:

- Expanding the model database through market analysis of similar records.
- Creation of an LSTM network model for individual companies.
- Finding the optimal capital–property structure.

# Bibliography

[1] Periklis Gogas and Theophilos Papadimitriou. “Machine learning in economics and finance”. In: *Computational Economics* 57 (2021), pp. 1–4.

[2] Halbert White. “Economic prediction using neural networks: The case of IBM daily stock returns”. In: *ICNN*. Vol. 2. 1988, pp. 451–458.

[3] *Giełda Papierów Wartościowych - archiwum*. 2021. URL: https://www.gpw.pl/statystyki-gpw (visited on 2023-11-28).

[4] Maryam Lotfian, Jens Ingensand, and Maria Antonia Brovelli. “The partnership of citizen science and machine learning: benefits, risks, and future challenges for engagement, data collection, and data quality”. In: *Sustainability* 13.14 (2021), p. 8087.

[5] Franco Modigliani and Merton H Miller. “The cost of capital, corporation finance and the theory of investment”. In: *The American economic review* 48.3 (1958), pp. 261–297.

[6] Faruk Ahmeti and Burim Prenaj. “A critical review of Modigliani and Miller’s theorem of capital structure”. In: *International Journal of Economics, Commerce and Management (IJECM)* 3.6 (2015).

[7] Sylwia Bętkowska. “Determinanty struktury kapitału w przedsiębiorstwie”. In: *Finanse, Rynki Finansowe, Ubezpieczenia* 82 (1) (2016), pp. 385–396.

[8] Frank E Grubbs. “Procedures for detecting outlying observations in samples”. In: *Technometrics* 11.1 (1969), pp. 1–21.

[9] Corporate Finance Institute. *Balance Sheet*. 2024. URL: https://corporatefinanceinstitute.com/resources/accounting/balance-sheet/ (visited on 2023-11-28).

[10] Investopedia. *Market Value*. 2024. URL: https://www.investopedia.com/terms/m/marketvalue.asp (visited on 2023-11-28).

[11] Text Analysis Pedagogy Institute. *Constellate*. 2023. URL: https://constellate.org/ (visited on 2023-06-14).

[12] Samuel Antwi, Ebenezer Fiifi Emire Atta Mills, and Xicang Zhao. “Capital structure and firm value: Empirical evidence from Ghana”. In: *International Journal of Business and Social Science* 3.22 (2012).

[13] Thi Ngoc Bui, Xuan Hung Nguyen, and Kieu Trang Pham. "The effect of capital structure on firm value: A study of companies listed on the Vietnamese stock market". In: *International Journal of Financial Studies* 11.3 (2023), p. 100.

[14] Maryam Alhani Fumani and Abdolkarim Moghadam. "The effect of capital structure on firm value, the rate of return on equity and earnings per share of listed companies in Tehran Stock Exchange". In: *Research journal of Finance and Accounting* 6.15 (2015), pp. 50–57.

[15] Varun Chandola, Arindam Banerjee, and Vipin Kumar. "Anomaly detection: A survey". In: *ACM computing surveys (CSUR)* 41.3 (2009), pp. 1–58.

[16] *Internet Crime Complaint Center.* 2019. URL: https://pdf.ic3.gov/2019_IC3Report.pdf (visited on 2023-11-28).

[17] Waleed Hilal, S Andrew Gadsden, and John Yawney. "Financial fraud: a review of anomaly detection techniques and recent advances". In: *Expert systems With applications* 193 (2022), p. 116429.

[18] Soumaya Ounacer et al. "Using Isolation Forest in anomaly detection: the case of credit card transactions". In: *Periodicals of Engineering and Natural Sciences* 6.2 (2018), pp. 394–400.

[19] Eugen Stripling et al. "Isolation-based conditional anomaly detection on mixed-attribute data to uncover workers' compensation fraud". In: *Decision Support Systems* 111 (2018), pp. 13–26.

[20] Zahra Kazemi and Houman Zarrabi. "Using deep networks for fraud detection in the credit card transactions". In: *2017 IEEE 4th International conference on knowledge-based engineering and innovation (KBEI)*. IEEE. 2017, pp. 0630–0633.

[21] Apapan Pumsirirat and Yan Liu. "Credit card fraud detection using deep learning based on auto-encoder and restricted boltzmann machine". In: *International Journal of advanced computer science and applications* 9.1 (2018).

[22] Markus M Breunig et al. "LOF: identifying density-based local outliers". In: *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*. 2000, pp. 93–104.

[23] Jian Tang et al. "Enhancing effectiveness of outlier detections for low density patterns". In: *Advances in Knowledge Discovery and Data Mining: 6th Pacific-Asia Conference, PAKDD 2002 Taipei, Taiwan, May 6–8, 2002 Proceedings 6*. Springer. 2002, pp. 535–548.

[24] Spiros Papadimitriou et al. "Loci: Fast outlier detection using the local correlation integral". In: *Proceedings 19th international conference on data engineering (Cat. No. 03CH37405)*. IEEE. 2003, pp. 315–326.

[25] Wen Jin et al. “Ranking outliers using symmetric neighborhood relationship”. In: *Advances in Knowledge Discovery and Data Mining: 10th Pacific-Asia Conference, PAKDD 2006, Singapore, April 9-12, 2006. Proceedings 10.* Springer. 2006, pp. 577–593.

[26] Hans-Peter Kriegel et al. “LoOP: local outlier probabilities”. In: *Proceedings of the 18th ACM conference on Information and knowledge management.* 2009, pp. 1649–1652.

[27] Mennatallah Amer and Markus Goldstein. “Nearest-neighbor and clustering based anomaly detection algorithms for rapidminer”. In: *Proc. of the 3rd RapidMiner Community Meeting and Conference (RCOMM 2012).* 2012, pp. 1–12.

[28] Zengyou He, Xiaofei Xu, and Shengchun Deng. “Discovering cluster-based local outliers”. In: *Pattern recognition letters* 24.9-10 (2003), pp. 1641–1650.

[29] Omar Alghushairy et al. “A review of local outlier factor algorithms for outlier detection in big data streams”. In: *Big Data and Cognitive Computing* 5.1 (2020), p. 1.

[30] Aleksandar Lazarevic et al. “A comparative study of anomaly detection schemes in network intrusion detection”. In: *Proceedings of the 2003 SIAM international conference on data mining.* SIAM. 2003, pp. 25–36.

[31] Sarah M Erfani et al. “High-dimensional and large-scale anomaly detection using a linear one-class SVM with deep learning”. In: *Pattern Recognition* 58 (2016), pp. 121–134.

[32] Tadaaki Hosaka. “Bankruptcy prediction using imaged financial ratios and convolutional neural networks”. In: *Expert systems with applications* 117 (2019), pp. 287–299.

[33] Jakub Horak, Jaromir Vrbka, and Petr Suler. “Support vector machine methods and artificial neural networks used for the development of bankruptcy prediction models and their comparison”. In: *Journal of Risk and Financial Management* 13.3 (2020), p. 60.

[34] Marek Vochozka, Jaromir Vrbka, and Petr Suler. “Bankruptcy or success? the effective prediction of a company’s financial development using LSTM”. In: *Sustainability* 12.18 (2020), p. 7529.

[35] Yu Lin et al. “Forecasting stock index price using the CEEMDAN-LSTM model”. In: *The North American Journal of Economics and Finance* 57 (2021), p. 101421.

[36] Seol-Hyun Noh. “Comparing the Performance of Corporate Bankruptcy Prediction Models Based on Imbalanced Financial Data”. In: *Sustainability* 15.6 (2023), p. 4794.

[37] Ian T Jolliffe and Jorge Cadima. “Principal component analysis: a review and recent developments”. In: *Philosophical transactions of the royal society A: Mathematical, Physical and Engineering Sciences* 374.2065 (2016), p. 20150202.

[38] Markus M Breunig et al. "LOF: identifying density-based local outliers". In: *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*. 2000, pp. 93–104.

[39] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. "Isolation forest". In: *2008 eighth ieee international conference on data mining*. IEEE. 2008, pp. 413–422.

[40] Corinna Cortes and Vladimir Naumovich Vapnik. "Support-Vector Networks". In: *Machine Learning* 20 (1995), pp. 273–297.

[41] Geoffrey E Hinton and Sam Roweis. "Stochastic neighbor embedding". In: *Advances in neural information processing systems* 15 (2002).

[42] Radosław Łazarz. *Zadanie 03 Pierwszy i prawdopodobnie ostatni raz w życiu trenujemy od podstaw konwolucyjną sieć neuronową*. 2022. URL: https://upel.agh.edu.pl/mod/assign/view.php?id=34288 (visited on 2024-05-08).

[43] IBM. *Logistic Regression*. IBM. Accessed 2024. URL: https://www.ibm.com/topics/logistic-regression (visited on 2023-11-28).

[44] Evelyn Fix. *Discriminatory analysis: nonparametric discrimination, consistency properties*. Vol. 1. USAF school of Aviation Medicine, 1985.