

Study of the Profitability of the Enterprise based on the Method of Machine Learning without a Teacher

Mariia Nazarkevych^{1,2}, Hanna Nazarkevych¹, Roman Moravskiy¹, Maryna Kostiak¹, and Oleksii Shevchuk³

¹ Lviv Polytechnic National University, 12 Stepan Bandera str., Lviv, 79013, Ukraine

² Lviv Ivan Franko National University, 1 Universytetska str., Lviv, 79000, Ukraine

³ Ukrainian Academy of Printing, 19 Pid Goloskom str., Lviv, 79020, Ukraine

Abstract

Machine learning methods, which are used in the framework of predicting the solution of methods of adaptive management of the enterprise's goods sales, have been analyzed. Conduct an analysis of input data obtained during the operation of one enterprise. With the help of input data, train and conduct machine learning simulations with K-Nearest Neighbors, Support Vector Machines classifiers. The most significant factors influencing the client's decision to purchase goods have been identified. The study proposed a business process scenario for solving the problem of increasing the company's profit based on machine learning technology. The performance of the proposed methods was verified on a test sample of data.

Keywords

Unsupervised learning methods, adaptive management, enterprise.

1. Introduction

In an era where organizations increasingly rely on data to run their business, the ability of different companies to manage and analyze data will have a significant impact on their performance. In this article, we develop a model to measure the current level of maturity of an enterprise's data and analytics systems to optimize them to maximize the potential of their data assets. Based on the three critical components of people, technology and process, we take the requirements of the enterprise as a starting point. After a comprehensive review of the current situation in the industry and the needs of the enterprise, a methodology for determining profitability was built [1].

Profitability [2] is the ratio of profit and costs, expressed as a percentage. Profitability is a relative indicator, and it is necessary for the analysis of economic and economic activity of any enterprise [3–5].

Let's consider the main indicators of the company's profitability:

1. Profitability of invested funds [6], which consists of the general level of profitability of the enterprise, which in turn is equal to the ratio of the gross profit of the enterprise to the total production cost.

2. Another indicator is the profitability of production assets [7], which in turn is equal to the ratio of the gross profit of the enterprise to the profitability of total assets.

Return on total assets [8] is the ratio of the company's gross profit to the average amount of assets on the company's balance sheet. 1. Return on equity (equity) [9] is the ratio of the company's net profit to the amount of equity.

2. And another indicator is the profitability of production [10], which is equal to the ratio of the total cost of goods sold to the volume of sales.

One of the fundamental prerequisites in the field of artificial intelligence is the assumption of the possibility of creating machines that are

CPITS-2022: Cybersecurity Providing in Information and Telecommunication Systems, October 13, 2022, Kyiv, Ukraine

EMAIL: mariia.a.nazarkevych@lpnu.ua (M. Nazarkevych); hanna.ya.nazarkevych@lpnu.ua (H. Nazarkevych);

roman.moravskiy.mnpzm2021@lpnu.ua (R. Moravskiy); kostiak.maryna@lpnu.ua (M. Kostiak); uad.sow@gmail.com (O. Shevchuk)

ORCID: 0000-0002-6528-9867 (M. Nazarkevych); 0000-0002-1413-630X (H. Nazarkevych); 0000-0002-6695-1920 (R. Moravskiy); 0000-0002-6667-7693 (M. Kostiak); 0000-0002-3200-7317 (O. Shevchuk)



© 2022 Copyright for this paper by its authors.
Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings (CEUR-WS.org)

capable of performing tasks that usually require the addition of human intelligence [11].

The artificial neural network is designed to simulate learning processes in the human brain. Artificial neural networks are designed in such a way that they can recognize basic patterns (regularities, stable relationships hidden in data) and learn from them [12]. They can be used to solve classification, regression, and data segmentation problems [13]. Before providing neural network data, they must be converted into numerical form. For example, data of various nature, including visual and textual data, time series, etc. Decisions have to be made about how tasks should be presented so that they are understandable to neural networks.

The practical implementation of material objects involves the solution of the corresponding problems of synthesis. At the same time, the mathematical model of the object should not only adequately describe the physical processes that ensure obtaining the necessary initial characteristics of this object, but also allow the optimization process itself to be implemented. The software currently used to solve similar problems is characterized by either narrow specialization, or has a universal calculation device and requires an unacceptably long calculation time for complex objects. One of the options for implementing fast and flexible methods for solving synthesis problems is the use of approximation properties of artificial neural networks. Most of the research related to artificial neural networks is focused on the solution of combinatorial optimization and forecasting problems, therefore, research on the application of neural networks is relevant.

2. Stages of Developing a Machine Learning Model

The process of developing a certain model of machine learning [14] consists of the following stages:

- Process of data preparation and presentation.
- Algorithm design process.
- Process of training the algorithm on the available data.
- Algorithm validation process on test data.

A training sample [15] is a set of examples that we show the system so that, based on them,

it uncovers a certain hidden regularity that is responsible for the distribution of data in the training sample. Thanks to the discovery of such a regularity, the system will be able to use it to effectively predict answers on the test sample.

A neural network [16] is a certain complex function that has a huge number of parameters. Each of these parameters (called weights) is adapted to approximate the function to the state of how the data in the test sample is distributed. Very often, the number of these parameters is much larger than the description of the test data.

3. Construction of Neural Networks

3.1. Mathematical Model

Perceptron is a mathematical model proposed by F. Rosenblatt, which is described by the transformation $R^n \rightarrow R$ using the formula

$$v = \sum_{i=1}^m \omega_{ij} x_i$$

where $j = 1, \dots, n$; ω_{ij} is the weight of the perceptron; x_i – value of input signals. After receiving the result, the activation function f is applied to the received value in v . The resulting value is compared with the trigger threshold θ and a decision is made.

Perceptron learning consists of finding weight coefficients. Let there be a set of pairs of vectors $(x^n, y^n), \alpha = 1, \dots, p$, which is called a training sample. We will call a neural network trained on a given training dataset if, when applying each vector x^n to the inputs of the network, we get the corresponding vector y^n at the outputs each time. Recognition, processing of text, human language, music, images, 3D objects, tabular data, object detection in photos and videos, even creation of texts, images and videos—all this and much more is included in the practical application of neural networks.

The learning method proposed by Rosenblatt [17] consists in iterative substitution of the weight matrix, successively reducing the error in the output vectors. The algorithm has several stages:

Step 1. Initial values of all neuron weights $W(t = 0)$ are placed randomly.

Step 2. If the input image x^n is loaded, the result will be the output image $\tilde{y}^n \neq y^n$.

Step 3. The error vector $\delta^n = (y^n - \hat{y}^n)$ is calculated, carried out by the network at the output. Then it happens that the change in the vector of weighting coefficients in the area of small errors must be proportional to the error at the output and equal to zero if the error is zero.

Step three. The vector of weights is modified by the formula:

$$W(t + \Delta T) = W(t) + \eta x^n \cdot (\delta^n)^T,$$

where $0 < \eta < 1$ is learning pace.

Step 4. Steps 1–3 are repeated for all vectors to be trained. One cycle of sequential representation of the entire sample is called an epoch. Training ends after several epochs: (a) when the iterations converge, that is, the vector stops changing, or when (b) or when the complete, rubberized absolute error over all vectors becomes smaller than some small value.

Deep learning is based on complex systems consisting of a huge number of neurons. One neural network can have billions of such structural units as neurons or perceptrons. Accordingly, there are many ways to structure them. And depending on how they will be connected, different architectures of neural networks can be distinguished.

3.2. Optimization of Neural Network Parameters

One of the simplest gradient descent optimization algorithms looks like this:

$$w_{p+1} = w_p - \eta \nabla \text{Cost}(w_p)$$

We also justify that the Nesterov algorithm will be the optimal algorithm for finding the minimum [18]

$$\Delta p + 1 = \gamma \cdot \Delta p + \eta \cdot \nabla \text{Cost}(w_p - \gamma \Delta p),$$

$$w_{p+1} = w_p - \Delta p + 1.$$

A kind of refining operation is used here to find the gradient. Therefore, the method is sometimes called the accelerated Nesterov gradient. However, despite the fact that the proven upper estimate of the convergence time for this algorithm is minimal, it has not found its practical application. The thing is that the top score does not always coincide with the average. And in the case of this algorithm, the average convergence time estimate is close to the upper estimate. Therefore, the algorithm mostly works slowly.

3.3. Creation of a Neural Network

Human learning continues hierarchically. In the neural network of our brain, this process is carried out in several stages, each of which is characterized by its own degree of learning. At some stages, simple things are taught, at others—more complex.

In order to simulate the human learning process, layers of neurons are used in the construction of artificial neural networks. The idea of these neurons is suggested by biological processes. Each layer of an artificial neural network is a set of independent neurons, each neuron of a certain layer is connected to the neurons of an adjacent layer.

3.4. Neural Network Training

If we are dealing with n-dimensional input data, then the input layer will consist of n neurons. If m different classes are distinguished among our training (training) data, then the output layer will consist of m neurons. The layers nested between the input and output layers are called hidden. A simple neural network consists of a pair of layers, and a deep neural network consists of many layers.

Consider the case when we want to use a neural network for data classification. The first step is to collect relevant training data and label it. Each neuron acts as a simple function, and the neural network is trained until the error is less than a certain set value. The difference between predicted and actual outputs is mostly used as an error. Based on how big the error is, the neural network corrects itself and retrain until it gets close to the solution.

3.5. Creating a Classifier based on a Perceptron

A perceptron is a single neuron that receives input data, performs calculations on it and outputs an output signal. The perceptron uses a simple function to make decisions. Suppose we are dealing with an N-dimensional input data point. The perceptron calculates a weighted sum of N numbers, and then adds a constant to them to get the original result. The ego constant is called the bias of the neuron. It is interesting to note that such simple perceptrons are used to design very complex deep neural networks.

3.6. Construction of a Single-Layer Neural Network

The capabilities of the perceptron are limited. You need to get a set of neurons to act as one and evaluate it to achieve a goal. Let's create a single-layer neural network consisting of independent neurons that influence the input data to obtain the output result.

3.7. Building a Multilayer Neural Network

To obtain higher accuracy, we must give more freedom to the neural network. This means that the neural network must have more than one layer to capture the underlying patterns that exist among the test data. Let's create a multilayer neural network that will ensure this.

A neural network can be used as a classifier. You can use a neural network like regressor [19, 20].

Let's define a multilayer neural network with two hidden layers. You can design a neural network in any other way. In this case, we will have 10 neurons in the first layer and six neurons in the second layer. Our task is to predict a single value, so the output layer will contain only one neuron.

4. Deployment of Machine Learning

We will use the creation of the model for the forecasting task.

To do this, we will help predict the probability that the enterprise will be profitable and will have profits in the future.

The dataset we used in this chapter is taken from this database: ML_Manufacture_Prifit_Companies.csv. This data set contains several independent predictors and one goal, to make the enterprise profitable. Its features are as follows: rating, change in rating, income, management assets, market value, percentage change in income, percentage change in profit, employees.

The data set contains 1001 records and all businesses are successful and profitable.

5. Downloading Data

For this example, the dataset was downloaded locally and named ML_Manufacture_Prifit_Companies.csv (Fig. 1). We observe the columns with the following data (Fig. 2). The first task for estimating a company's profit is to clean the data so that there are no missing or erroneous values (Fig. 3 and 4) [21–23].

index	rank	name	revenues	revenue_percent_change	profits	profits_percent_change	assets	market_value	change_in_rank	employees
0	1	Walmart	\$572,754	2.40%	\$13,673	1.20%	\$244,860	\$409,795	-	2,300,000
1	2	Amazon	\$469,822	21.70%	\$33,364	56.40%	\$420,549	\$1,658,807.30	-	1,608,000
2	3	Apple	\$365,817	33.30%	\$94,680	64.90%	\$351,002	\$2,849,537.60	-	154,000
3	4	CVS Health	\$292,111	8.70%	\$7,910	10.20%	\$232,999	\$132,839.20	-	258,000
4	5	UnitedHealth Group	\$287,597	11.80%	\$17,285	12.20%	\$212,206	\$479,830.30	-	350,000
5	6	Exxon Mobil	\$285,640	57.40%	\$23,040	-	\$338,923	\$349,652.40	4	63,000
6	7	Berkshire Hathaway	\$276,094	12.50%	\$89,795	111.20%	\$958,784	\$779,542.30	-1	372,000
7	8	Alphabet	\$257,637	41.20%	\$76,033	88.80%	\$359,268	\$1,842,326.10	1	156,500
8	9	McKesson	\$238,228	3.10%	(\$4,539)	-604.30%	\$65,015	\$45,857.80	-2	67,500
9	10	AmerisourceBergen	\$213,988.80	12.70%	\$1,539.90	-	\$57,337.80	\$32,355.70	-2	40,000
10	11	Cosco Wholesale	\$195,929	17.50%	\$5,007	25.10%	\$59,268	\$255,230.70	1	288,000
11	12	Cigna	\$174,078	8.50%	\$5,365	-36.60%	\$154,889	\$76,286.30	1	72,963
12	13	AT&T	\$168,864	-1.70%	\$20,081	-	\$551,622	\$169,262.40	-2	202,600
13	14	Microsoft	\$168,088	17.50%	\$61,271	38.40%	\$333,779	\$2,311,358.90	1	181,000
14	15	Cardinal Health	\$162,467	6.20%	\$611	-	\$44,453	\$15,709.40	-1	46,827
15	16	Chevron	\$162,465	71.60%	\$15,625	-	\$239,535	\$317,120.10	11	42,595
16	17	Home Depot	\$151,157	14.40%	\$16,433	27.70%	\$71,876	\$309,312.60	1	490,600
17	18	Walgreens Boots Alliance	\$148,579	6.50%	\$2,542	457.50%	\$81,285	\$38,671.10	-2	258,500
18	19	Marathon Petroleum	\$141,032	58.50%	\$9,738	-	\$85,373	\$47,758.10	13	17,700
19	20	Elevance Health	\$138,639	13.80%	\$6,104	33.50%	\$97,460	\$118,533.50	3	98,200
20	21	Kroger	\$137,888	4.10%	\$1,655	-36%	\$49,086	\$41,496.20	-4	420,000
21	22	Ford Motor	\$136,341	7.20%	\$17,937	-	\$257,035	\$67,958.20	-1	183,000
22	23	Verizon Communications	\$133,613	4.10%	\$22,065	24%	\$366,596	\$213,837.10	-3	118,400
23	24	JPMorgan Chase	\$127,202	-1.80%	\$48,334	65.90%	\$3,743,567	\$402,526.90	-5	271,025
24	25	General Motors	\$127,004	3.70%	\$10,019	55.90%	\$244,718	\$63,555.20	-3	157,000

Figure 1: Download dataset

```

    'profits_percent_change',
    'employees'
]

df = clean_data(df, columns_to_clean).fillna(0)

df['rank'] = df['rank'].astype(np.int64)
df['change_in_rank'] = df['change_in_rank'].astype(np.int64)
df['employees'] = df['employees'].astype(np.int64)

df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 10 columns):
#   Column              Non-Null Count  Dtype
---  ---
0   rank                 1000 non-null   int64
1   name                 1000 non-null   object
2   revenues             1000 non-null   float64
3   revenue_percent_change 1000 non-null   float64
4   profits              1000 non-null   float64
5   profits_percent_change 1000 non-null   float64
6   assets               1000 non-null   float64
7   market_value         1000 non-null   float64
8   change_in_rank       1000 non-null   int64
9   employees            1000 non-null   int64
dtypes: float64(6), int64(3), object(1)
memory usage: 78.2+ KB

```

Figure 2: Data cleaning

index	rank	name	revenues	revenue_percent_change	profits	profits_percent_change	assets	market_value	change_in_rank	employees	
0	1	Walmart	572754.0	2.4	13673.0	1.2	244860.0	409795.0	0	2300000	
1	2	Amazon	469822.0		21.7	33364.0	56.4	420549.0	1658807.3	0	1608000
2	3	Apple	365817.0		33.3	94680.0	64.9	351002.0	2849537.6	0	154000
3	4	CVS Health	292111.0		8.7	7910.0	10.2	232999.0	132839.2	0	258000
4	5	UnitedHealth Group	287597.0		11.8	17285.0	12.2	212206.0	479830.3	0	350000
5	6	Exxon Mobil	285640.0		57.4	23040.0	0.0	338923.0	349652.4	4	63000
6	7	Berkshire Hathaway	276094.0		12.5	89795.0	111.2	958784.0	779542.3	1	372000
7	8	Alphabet	257637.0		41.2	76033.0	88.8	359268.0	1842326.1	1	156500
8	9	McKesson	238228.0		3.1	4539.0	604.3	65015.0	45857.8	2	67500
9	10	AmerisourceBergen	213988.8		12.7	1539.9	0.0	57337.8	32355.7	2	40000
10	11	Costco Wholesale	195929.0		17.5	5007.0	25.1	59268.0	255230.7	1	288000
11	12	Cigna	174078.0		8.5	5365.0	36.6	154889.0	76286.3	1	72963
12	13	AT&T	168864.0		1.7	20081.0	0.0	551622.0	169262.4	2	202600
13	14	Microsoft	168088.0		17.5	61271.0	38.4	333779.0	2311358.9	1	181000
14	15	Cardinal Health	162467.0		6.2	611.0	0.0	44453.0	15709.4	1	46827
15	16	Chevron	162465.0		71.6	15625.0	0.0	239535.0	317120.1	11	42595
16	17	Home Depot	151157.0		14.4	16433.0	27.7	71876.0	309312.6	1	490600
17	18	Walgreens Boots Alliance	148579.0		6.5	2542.0	457.5	81285.0	38671.1	2	258500
18	19	Marathon Petroleum	141032.0		58.5	9738.0	0.0	85373.0	47758.1	13	17700
19	20	Elevance Health	138639.0		13.8	6104.0	33.5	97460.0	118533.5	3	98200
20	21	Kroger	137888.0		4.1	1655.0	36.0	49086.0	41496.2	4	420000
21	22	Ford Motor	136341.0		7.2	17937.0	0.0	257035.0	67958.2	1	183000
22	23	Verizon Communications	133613.0		4.1	22065.0	24.0	366596.0	213837.1	3	118400
23	24	JPMorgan Chase	127202.0		1.8	48334.0	65.9	3743567.0	402526.9	5	271025
24	25	General Motors	127004.0		3.7	10019.0	55.9	244718.0	63555.2	3	157000

Figure 3: Cleaned and transformed data

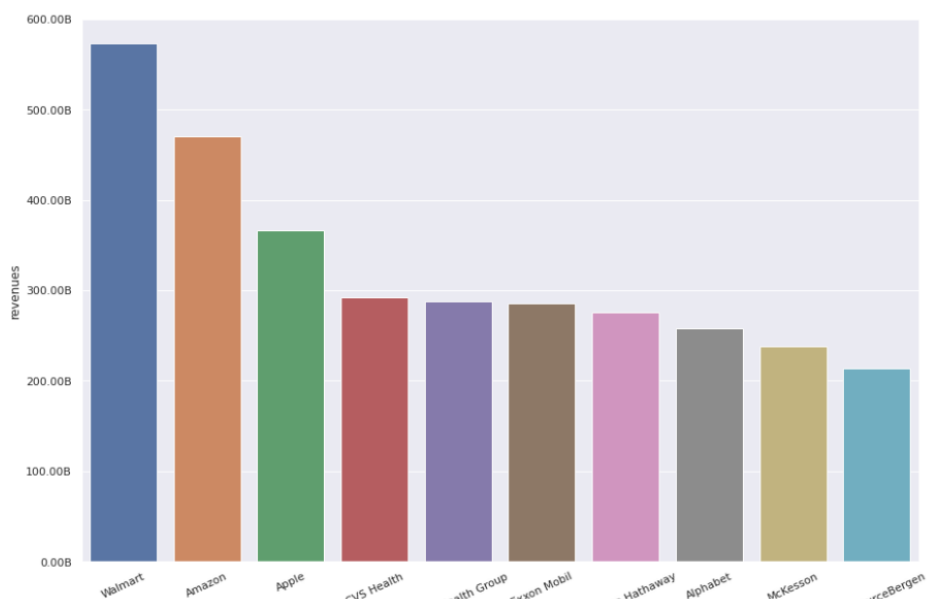


Figure 4: Visualization of the top 10 companies by revenue

6. Study of the Relationship Between Features

The next step is to study how different independent characteristics affect the result (which characteristics affect the profitability of the enterprise).

The `corr()` function calculates the pairwise correlation of columns. For example, the following result shows that the level of management assets has little relationship with the market value of the enterprise 0.127, but it has a significant relationship with the profitability of the firm 0.49.

We find out which functions significantly affect the result.

7. Construction of a Graph of the Relationship between Features

The following code snippet uses the `matshow()` function to plot the results returned by the `corr()` function as a matrix. At the same time, various correlation coefficients are also displayed in the matrix:

Fig. 5 shows the matrix. Cubes with colors closest to black represent the highest correlation coefficients, and those closest to blue represent the lowest correlation coefficients.

Another way to construct a correlation matrix is to use Seaborn's `heatmap()` function as follows: Fig. 5 shows a heat map created using the Seaborn library (Fig. 6).

Let's single out four signs at enterprises that have the highest correlation with the result (Fig. 7–9).

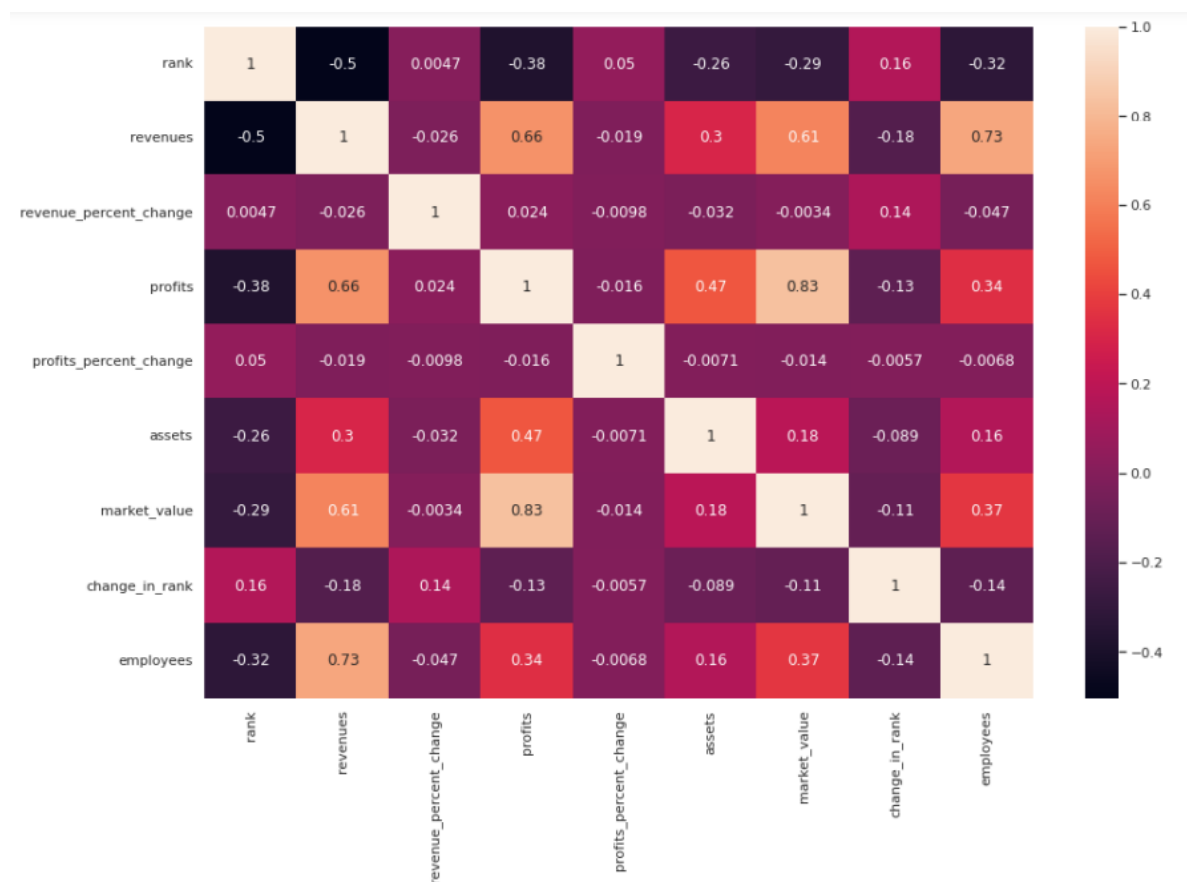


Figure 5: Matrix showing various correlation factors

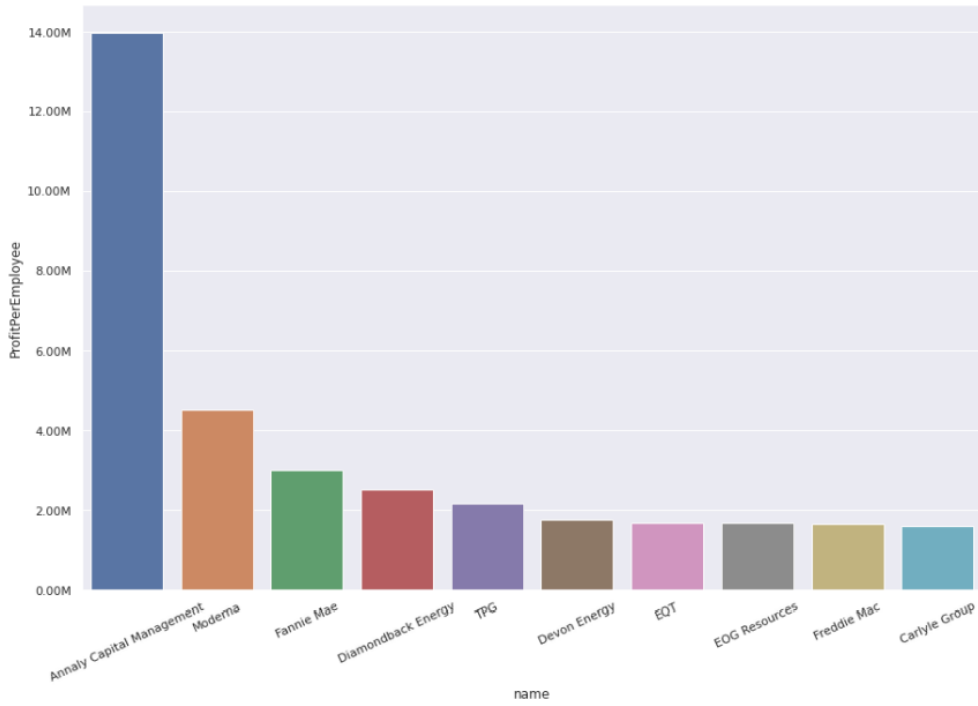


Figure 6: Visualization of the top 10 companies with the largest profit in relation to the number of employees

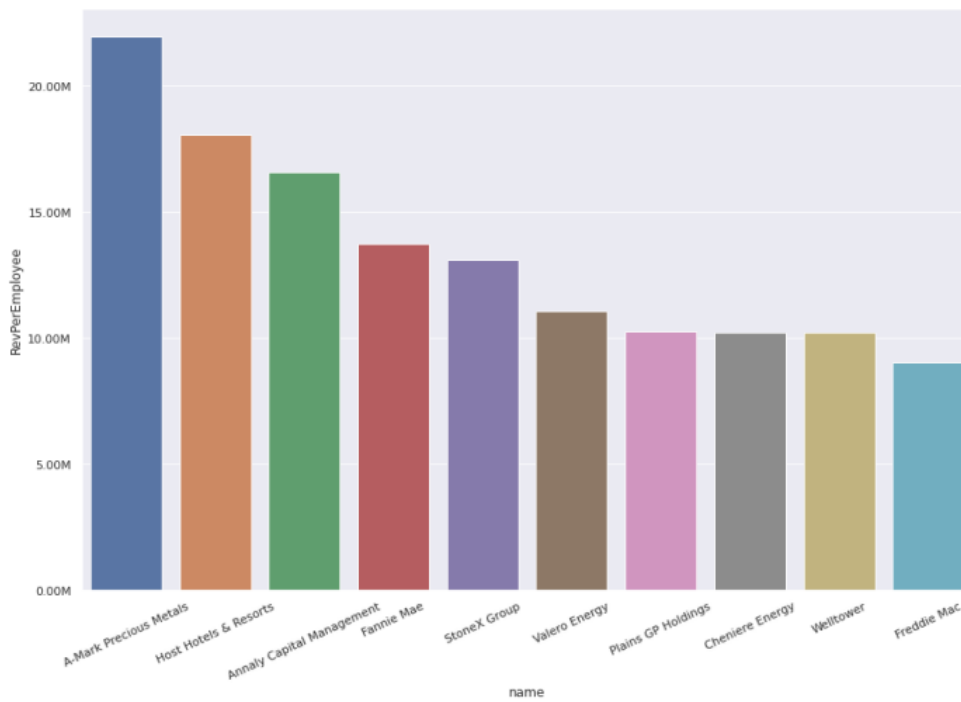


Figure 7: Visualization of the top 10 companies with the largest income per number of employees

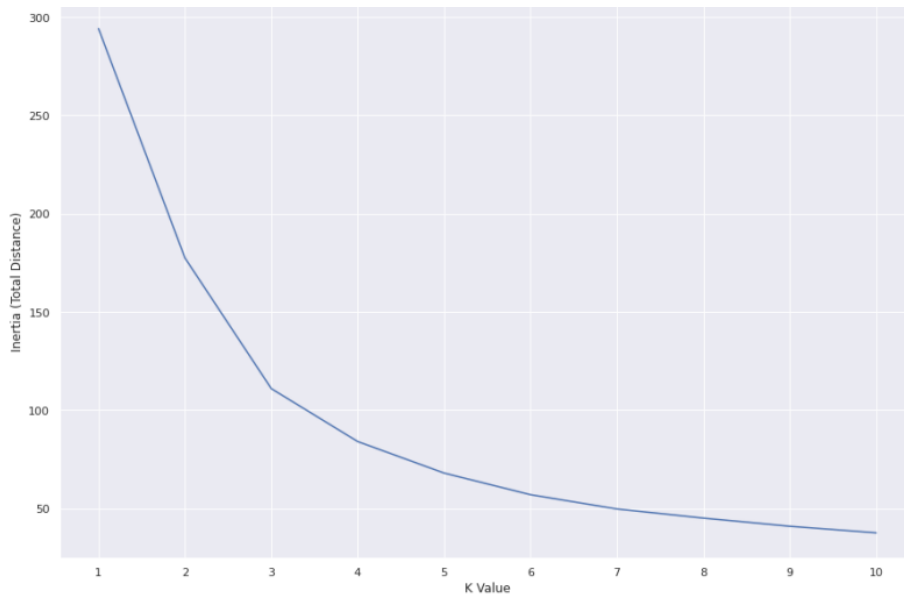


Figure 8: Graph of dependence of the Inertia metric on the number of clusters



Figure 9: Found clusters of companies

8. Evaluation of Algorithms

We will evaluate several algorithms to find the most optimal one and the one that will provide the best performance. Therefore, we will use the following methods:

- Logistic regression.
- K-Nearest Neighbors (KNN).
- Support vector machines (SVM) using linear kernels and RBF kernels.

8.1. Logistic Regression

For the first method, we will use logistic regression. Instead of splitting the dataset into training and testing sets, we will use 10-fold cross-validation to get the average score of the algorithm used.

8.2. K-Nearest Neighbours

The next method we will use is K-Nearest Neighbors (KNN). In addition to using 10-fold cross-validation to average the algorithm,

different values of k should be tried to find the optimal one so that the best accuracy can be obtained.

8.3. Support Vector Machines

Another method we will use is Support Vector Machine (SVM). We will use two types of kernels for SVM: linear and RBF.

9. Learning and Saving the Model

Since the most efficient algorithm for our data set is KNN with $k = 3$, we can now continue to train the model using the entire data set: After training the model, we need to save

the files so that we can get the model for prediction later. The trained model is now saved to a file.

After loading the model, let's make some predictions:

The output prints the word "Profitable Enterprise" if the return value of the prediction is 1; otherwise, the word "Unprofitable enterprise" is printed. It is also necessary to obtain the probability of the forecast in order to obtain the probability in percent (Fig. 10–12).

The printed probabilities show that the probability of the outcome is 0 and the probability of the outcome is 1. The prediction is based on the prediction with the highest probability, and we take that probability and convert it to a confidence percentage.

index	rank	name	revenues	revenue_percent_change	profits	profits_percent_change	assets	market_value	change_in_rank	employees	kmeans_label
6	7	Berkshire Hathaway	276094.0	12.5	89795.0	111.2	958784.0	779542.3	1	372000	0
9	10	AmerisourceBergen	213988.8	12.7	1539.9	0.0	57337.8	32355.7	2	40000	0
11	12	Cigna	174078.0	8.5	5365.0	36.6	154889.0	76286.3	1	72963	0
12	13	AT&T	168864.0	1.7	20081.0	0.0	551622.0	169262.4	2	202600	0
18	19	Marathon Petroleum	141032.0	58.5	9738.0	0.0	85373.0	47758.1	13	17700	0
21	22	Ford Motor	136341.0	7.2	17937.0	0.0	257035.0	67958.2	1	183000	0
22	23	Verizon Communications	133613.0	4.1	22065.0	24.0	366596.0	213837.1	3	118400	0
23	24	JPMorgan Chase	127202.0	1.8	48334.0	65.9	3743567.0	402526.9	5	271025	0
24	25	General Motors	127004.0	3.7	10019.0	55.9	244718.0	63555.2	3	157000	0
27	28	Comcast	116385.0	12.4	14159.0	34.4	275905.0	212245.8	2	189000	0
28	29	Phillips 66	114852.0	75.4	1317.0	0.0	55594.0	41561.0	19	14000	0
29	30	Valero Energy	108332.0	80.2	930.0	0.0	57888.0	41572.2	23	9804	0

Figure 10: Example of data from companies from cluster 0

index	rank	name	revenues	revenue_percent_change	profits	profits_percent_change	assets	market_value	change_in_rank	employees	kmeans_label
0	1	Walmart	572754.0	2.4	13673.0	1.2	244860.0	409795.0	0	2300000	1
3	4	CVS Health	292111.0	8.7	7910.0	10.2	232999.0	132839.2	0	258000	1
8	9	McKesson	238228.0	3.1	4539.0	604.3	65015.0	45857.8	2	67500	1
10	11	Costco Wholesale	195929.0	17.5	5007.0	25.1	59268.0	255230.7	1	288000	1
14	15	Cardinal Health	162467.0	6.2	611.0	0.0	44453.0	15709.4	1	46827	1
16	17	Home Depot	151157.0	14.4	16433.0	27.7	71876.0	309312.6	1	496000	1
17	18	Walgreens Boots Alliance	148579.0	6.5	2542.0	457.5	81285.0	38671.1	2	258500	1
20	21	Kroger	137888.0	4.1	1655.0	36.0	49086.0	41496.2	4	420000	1
25	26	Centene	125982.0	13.4	1347.0	25.5	78375.0	49072.2	2	72500	1
30	31	Dell Technologies	106995.0	13.6	5563.0	71.2	92735.0	38164.4	3	133000	1
31	32	Target	106005.0	13.3	6946.0	59.0	53811.0	98134.4	2	450000	1
33	34	UPS	97287.0	15.0	12890.0	859.8	69405.0	186816.6	1	400945	1
34	35	Lowe's	96250.0	7.4	8442.0	44.7	44640.0	133761.1	4	270000	1

Figure 11: Example of data from companies from cluster 1

index	rank	name	revenues	revenue_percent_change	profits	profits_percent_change	assets	market_value	change_in_rank	employees	kmeans_label
1	2	Amazon	469822.0	21.7	33364.0	56.4	420549.0	1658807.3	0	1608000	2
2	3	Apple	365817.0	33.3	94680.0	64.9	351002.0	2849537.6	0	154000	2
4	5	UnitedHealth Group	287597.0	11.8	17285.0	12.2	212206.0	479830.3	0	350000	2
5	6	Exxon Mobil	285640.0	57.4	23040.0	0.0	338923.0	349652.4	4	63000	2
7	8	Alphabet	257637.0	41.2	76033.0	88.8	359268.0	1842326.1	1	156500	2
13	14	Microsoft	168088.0	17.5	61271.0	38.4	333779.0	2311358.9	1	181000	2
15	16	Chevron	162465.0	71.6	15625.0	0.0	239535.0	317120.1	11	42595	2
19	20	Elevance Health	138639.0	13.8	6104.0	33.5	97460.0	118533.5	3	98200	2
26	27	Meta Platforms	117929.0	37.2	39370.0	35.1	165987.0	605251.0	7	71970	2
36	37	Johnson & Johnson	93775.0	13.6	20878.0	41.9	182018.0	466046.7	1	141700	2
42	43	Pfizer	81288.0	94.0	21979.0	128.6	181476.0	292385.3	34	79000	2
45	46	Intel	79024.0	1.5	19868.0	4.9	168406.0	202635.8	6	121100	2
46	47	Procter & Gamble	76118.0	7.3	14306.0	9.8	119307.0	366271.6	4	101000	2
52	53	Walt Disney	67418.0	3.1	1995.0	0.0	203609.0	249718.1	3	171000	2
54	55	Lockheed Martin	67044.0	2.5	6315.0	7.6	50873.0	117648.2	6	114000	2

Figure 12: Example of data from companies from cluster 2

10. Conclusions

In this study, a dataset of top 1,000 profitable companies was selected and machine learning was implemented without a teacher. Companies were divided into three clusters based on revenue, earnings, current assets, market value and number of employees. Clustering was implemented based on the KMeans algorithm. The Inertia metric was also used to determine the optimal number of clusters. Inertia shows us the sum of the distances to each center of mass. If the total distance is large, it means that the points are far apart and may be less similar to each other. In this case, one can continue to evaluate larger values of K to see if the overall distance can be reduced. However, it is not always the smartest idea to reduce the distance. Using the elbow (bend) method, we can choose the value of the number of clusters 2 or 3 (the first or second obvious bend).

11. References

- [1] T. Zhang, et al., Enterprise Digital Transformation and Production Efficiency: Mechanism Analysis and Empirical Research. *Economic Research-Ekonomska Istraživanja*, vol. 35, no. 1, 2022, pp. 2781–2792.
- [2] S. Jahan, et al., Modeling Profitability-Influencing Risk Factors for Construction Projects: A System Dynamics Approach, *Buildings*, vol. 12, no. 6, 2022, pp. 701.
- [3] H. Hulak, et al. Formation of Requirements for the Electronic Record-Book in Guaranteed Information Systems of Distance Learning, in *Workshop on Cybersecurity Providing in Information and Telecommunication Systems, CPITS 2021*, vol. 2923, 2021, pp. 137–142.
- [4] V. Buriachok, V. Sokolov, Implementation of Active Learning in the Master's Program on Cybersecurity, in *Advances in Computer Science for Engineering and Education II*, 2019, vol. 938, 2020, pp. 610–624. doi: 10.1007/978-3-030-16621-2_57.
- [5] Z. B. Hu, et al., Authentication System by Human Brainwaves Using Machine Learning and Artificial Intelligence, in *Advances in Computer Science for Engineering and Education IV*, 2021, pp. 374–388. doi: 10.1007/978-3-030-80472-5_31.
- [6] A. I. Kato, C. P. E. Germinah, Empirical Examination of Relationship between Venture Capital Financing and Profitability of Portfolio Companies in Uganda, *Journal of Innovation and Entrepreneurship*, vol. 11, no. 1, 2022, pp. 1–18.
- [7] E. Filatov, Analysis of Profitability of Production of Enterprises in the Field of Transportation and Storage of the Irkutsk Region. *Transportation Research Procedia*, vol. 63, 2022, pp. 518–524.
- [8] H. Habibniya, et al., Impact of Capital Structure on Profitability: Panel Data Evidence of the Telecom Industry in the United States, *Risks*, vol. 10, no. 8, 2022, pp. 157.
- [9] V. V. Stipic, V. Ruzic, Panel VAR Analysis of the Interdependence of Capital Structure and Profitability, *Economic and Social Development: Book of Proceedings*, 2022, pp. 177–187.
- [10] E. A. Virtanen, et al., Balancing Profitability of Energy Production, Societal Impacts and Biodiversity in Offshore Wind Farm Design, *Renewable and Sustainable Energy Reviews*, vol. 158, 2022, pp. 112087.
- [11] X. Chen, P. Geyer, Machine Assistance in Energy-Efficient Building Design: A Predictive Framework Toward Dynamic Interaction with Human Decision-Making under Uncertainty, *Applied Energy*, vol. 307, 2022, pp. 118240.
- [12] G. V. S. Bhagya Raj, K. K. Dash, Comprehensive Study on Applications of Artificial Neural Network in Food Process Modeling, *Critical Reviews in Food Science and Nutrition*, vol. 62, no. 10, 2022, pp. 2756–2783.
- [13] F. Moosavi, et al., Application of Machine Learning Tools for Long-Term Diagnostic Feature Data Segmentation, *Applied Sciences*, vol. 12, no. 13, 2022, pp. 6766.
- [14] J. Siebert, et al., Construction of a Quality Model for Machine Learning Systems, *Software Quality Journal*, vol. 30, no. 2, 2022, pp. 307–335.
- [15] C. Zhang, et al., Mapping Irrigated Croplands in China using a Synergetic Training Sample Generating Method, Machine Learning Classifier, and Google Earth Engine, *International Journal of*

- Applied Earth Observation and Geoinformation, vol. 112, 2022, pp. 102888.
- [16] O. I. Abiodun, et al., State-of-the-Art in Artificial Neural Network Applications: A Survey, *Heliyon*, vol. 4, no. 11, 2018, pp. e00938.
- [17] E. Kussul, et al., Rosenblatt Perceptrons for Handwritten Digit Recognition, in *International Joint Conference on Neural Networks*, vol. 2, 2001, pp. 1516–1520.
- [18] Q. Lü, et al., A Nesterov-Like Gradient Tracking Algorithm for Distributed Optimization over Directed Networks, *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 10, 2020, pp. 6258–6270.
- [19] M. Medykovskyy, et al., Methods of Protection Document Formed from Latent Element Located by Fractals, in *X International Scientific and Technical Conference “Computer Sciences and Information Technologies,”* 2015, pp. 70–72.
- [20] M. Nazarkevych, et al., Application Perfected Wave Tracing Algorithm, in *IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON)*, 2017, pp. 1011–1014.
- [21] M. Nazarkevych, The Ateb-Gabor Filter for Fingerprinting, in *International Conference on Computer Science and Information Technology*, 2019, pp. 247–255. doi: 10.1007/978-3-030-33695-0_18.
- [22] M. Logoyda, et al., Identification of Biometric Images using Latent Elements. *CEUR Workshop Proceedings*, 2019.
- [23] M. Nazarkevych, et al., The Ateb-Gabor Filter for Fingerprinting, in *Conference on Computer Science and Information Technologies*, 2019, pp. 247–255.