IDX SHARE CLASSIFICATION TO THE LISTING BOARD BASED ON 5 MAIN FACTORS

1st Arya Jayavardhana
Majoring in Information Systems, Faculty of Technology and Information
Universitas Multimedia Nusantara Jl. Scientia Boulevard, Gading Serpong, Tangerang, Banten – 15811 Indonesia
arya.jayavardhana@student.umn.ac.id

2nd Leonardo
Majoring in Information Systems, Faculty of Technology and Information
Universitas Multimedia Nusantara Jl. Scientia Boulevard, Gading Serpong, Tangerang, Banten – 15811 Indonesia
leonardo5@student.umn.ac.id

3rd Steven Marcelino Tandiono
Majoring in Information Systems, Faculty of Technology and Information
Universitas Multimedia Nusantara Jl. Scientia Boulevard, Gading Serpong, Tangerang, Banten – 15811 Indonesia
steven.marcelino@student.umn.ac.id

4th Vinka Bella
Majoring in Information Systems, Faculty of Technology and Information
Universitas Multimedia Nusantara Jl. Scientia Boulevard, Gading Serpong, Tangerang, Banten – 15811 Indonesia
vinka.bella@student.umn.ac.id

Abstract.

Predicting stock market trends has become crucial in today's era, where stock developments have shown significant growth. Predictions are necessary for making informed investment decisions. Machine learning algorithms assist in stock prediction. Stock predictions are made by classifying IDX stocks into listing boards based on five key factors. The Decision Tree and K-Nearest Neighbors (KNN) machine learning algorithms are used, and both algorithms are evaluated to determine the best performance and results. This prediction is beneficial in identifying important factors and enhancing the understanding of stock determinants in the listing boards. It improves stock exchange efficiency and aids investors in making informed decisions regarding stock investments.The data used was obtained from the official IDX website, which will undergo several stages and processes. Then, it will proceed to the modeling stage, and the obtained results will be evaluated. The results showed that the Decision Tree algorithm has slightly more accurate results compared to the K-Nearest Neighbors (KNN) algorithm. Decision Tree has an accuracy of 76% and KNN is 73%.

Keywords: Classification, Decision Tree, Listing Board, Machine Learning, KNN

I. Introduction

Nowadays, stocks continue to experience significant developments. Shares describe the ownership of a person, in holding shares of a particular company. In predicting stock prices, technologies such as machine learning, big data analytics and other language processing. These algorithms can help analyze large amounts of data in real-time and make predictions based on patterns and trends. One of the most popular methods used to predict stocks is machine learning algorithms. This algorithm studies historical data, and based on the patterns it identifies, it can make predictions about future stock prices. This algorithm is trained on various data sources, including news articles, financial reports, market trends, and other related data [1].

When a company decides to go public and offer its shares, it can list its shares on a stock exchange such as IDX. IDX (Indonesia Stock Exchange) is the main stock exchange in Indonesia which is responsible for trading markets such as stocks, bonds and mutual funds. IDX provides a platform for companies to list their shares and for investors to buy and sell these shares.

Stock market trend prediction is considered as an important task because predicting stock prices successfully can generate attractive profits by making the right decisions. There are several prediction techniques designed to help predict stocks, IDX (Indonesia Stock Exchange) uses a classification system on the listing board to categorize companies based on several factors. The purpose of this is to provide a way for investors to identify companies that are suitable for the investment objectives they want to do and see the level of risk tolerance.

In this case, this project was carried out to find out whether classifying shares into a listing board can produce fast and accurate results. The IDX stock classification system is designed to provide a way for investors to assess potential returns and risks. By understanding the different classes of stocks and the factors used to classify companies, investors can make more informed investment decisions.

In this study, the authors have determined the formulation of the problem including 1)Which factors are most influential in classifying shares to the listing board. 2) Are the 5 factors capable of classifying shares to the listing board. 3) Which algorithm has better performance and results.

The purpose of research on the topic of IDX stock classification to the Listing Board is based on 5 main factors, namely 1) Determine or identify the determinants that influence the classification of shares to the listing board. 2) Determine whether these 5 factors are sufficient to classify stocks. 3) Define an algorithm that can produce better performance with see accuracy.. This research is expected to provide theoretical benefits, namely 1) Identify important factors in stock classification. 2) Increase understanding of the determinants of shares into the listing board. and practical benefits, namely 1) Increasing the efficiency of the stock exchange in classifying shares into 3 listing board classifications. 2) Assisting investors in making stock investment decisions.

As for classifying shares into the listing board, researchers used a comparison of two machine learning models, namely K-Nearest Neighbors (KNN) and Decision Trees. The use of the KNN model is considered feasible for predicting the price of the stock value based on the variables that influence it [2]. Then for the comparison of the two models, this refers to research previously used on the same model. Where is the research but aims to predict student performance by comparing models classification. The results of the research show that the KNN model has the same accuracy both by 93% and for the decision tree model by 92% [3]

II. Literature review

A. Normalization: MinMax Normalization

Normalization is a method of preprocessing where the process of normalization is changing data on existing attributes or features to a smaller value or uniform scale. The purpose of applying normalization is so that the attributes in the dataset do not dominate each other, which means that the data on the attributes have a uniform value scale. One method of doing normalization is min-max normalization. min-max normalization is a method that transforms data on features into a range of values from 0 to 1 [4]. Here is the formula for min-max normalization:

$$x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

Fig 1. MinMax Normalization Formula

B. Standardization: Standard Scaler

Standardization is a technique in the preprocessing stage that converts values into a uniform range. The purpose of implementing standardization is so that the existing data on each attribute is uniform so that the data is consistent or the range in the data is not too far away. One of the standardization methods is the standard scaler. The standard scaler is a method that removes the average and performs a range of unit variance [5]. The following is the formula of the standard scaler

$$Z = \frac{x_i - \overline{x}}{\sigma}$$

Fig 2. Standard scaler Formula

C. Z-score

Z-score analysis is a technique that can identify or detect outliers in the data. Identifying outliers is very suitable to be applied to eliminate noise in data. The Z-score is a statistical measure in standard deviation units that can display the farthest value of a data from the average [6]. The formula for dealing with outliers using the z-score can be seen as follows:

$$Z = \frac{x - \mu}{\sigma}$$

Fig 3. Z-score Formula

D. Resampling

The resampling method is useful for redistributing training data from different classes in the dataset. The purpose of using this method is to balance unbalanced classes and can even improve modeling performance. The types of resampling methods can be divided into under-sampling, over-sampling, and hybrid-sampling [7].

The resampling method used to overcome data imbalances is the Synthetic Minority Oversampling Technique + Tomek Link (SMOTETomek). This method is a combination of the SMOTE oversampling technique to increase the sample and Tomek Link to clean up overlapping samples [8]. The benefit of using this combined or hybrid method is that it is effective in balancing data and improving the minority class and the majority class simultaneously.

E. K-Nearest Neighbors

K-Nearest Neighbor (KNN) is a method that can be used to classify new test data according to learning data, which is based on the majority of K values of neighbors with the closest distance [9]. In short, the goal of KNN modeling is to classify new test data based on the majority of the K nearest neighbors of the training data.

In determining the best K value in modeling depends on the nature of the data used. A high K value reduces the effect of noise or outliers in the data, but also creates the risk of over-generalization or underfitting which makes the classification process blurry. Whereas a low K value captures better details, it also creates a high noise or outlier effect on the data.

In the KNN algorithm to find the nearest or farthest neighbors, in general it can be calculated using the Euclidean distance. The formula can be seen in Figure 4, namely [10]:

$$d(x,y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

Fig 4. KNN Formula

There are a series of steps to calculate the algorithm or method from K-Nearest Neighbors, namely 1) Determine the K parameter value or the number of nearest neighbours. 2) Calculate the distance between the training data and test data using the Euclidean distance formula. 3) Sort the calculation results in ascending order. 4) Classify the new test data based on the most labels of the K value [11].

F. Decision Tree Classifier

Decision Tree or decision tree is a method used for classification and regression. The Decision tree method is a decision-making tool using a decision model that has a tree-like structure. Like a tree, a decision tree has three main parts, namely root nodes, branches and leaf nodes. Where the root node represents a feature or attribute, each branch represents a decision rule, and the leaf node represents a label for each class [12]. The aim of this method is to predict targets based on decision rules formed from datasets and related features [13].

In the Decision Tree algorithm, it is necessary to determine the best attribute for the root and leaf nodes. Where this problem can be solved by using attribute selection calculation techniques, such as entropy and information gain. Entropy is calculating the randomness or impurity in the data, the higher the entropy, the more random the data. The formula for calculating entropy looks like in Figure 5, namely:

$$Entropy(S) = \sum_{i=1}^{c} -P_i \log_2 P_i$$

Fig 5. Entropy Formula

Meanwhile, information gain is a measure of entropy change to determine the best separation. Simply put, Information gain calculates how much information an attribute or feature can provide about a class. The information gain calculation can be done with the following equation:

$$Information\ Gain = E_k - (P_m\ E_m)$$

Fig 6. Information Gain Formula

There are stages in the Decision Tree method that can be described, namely 1) Determine the attribute as the root node. 2) Set a branch for each attribute value at the root node. 3) Dividing cases into branches according to the value of the attribute. 4) Do recursion for each branch until it has the same class. 5) Determination of attributes as nodes is determined based on the highest gain value of each existing attribute [14].

G. Evaluation Matrix

Evaluation Matrix is a tool for measuring or evaluating the performance of machine learning models built to process data [15]. The benefit of using an evaluation matrix is to be able to measure the accuracy or success of a model that has been made. In this project the evaluation matrix measurement uses the confusion matrix.

Confusion Matrix is an evaluation matrix method that is used to measure the value of accuracy, precision, and recall of machine learning models that are made [16]. In the Confusion Matrix there are 4 components namely True Positive (TP), True Negative (TN), False Positive (FP), False Negative (FN).

Testing a model based on the results of the evaluation is calculated using the confusion matrix technique. Measurement of the success rate of the model is obtained from the accuracy, precision, and recall methods [17].

- *Accuracy*
  The following is the formula for the accuracy that predicts the model built in both the positive and negative classes [18] :

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \times 100\%$$

Fig 7. Accuracy Formula

- *Precision*
  The precision formula calculates and predicts the ratio of true positives compared to the total true positive data. Precision formula as in figure 8:

$$Precision = \frac{TP}{TP+FP} \times 100\%$$

Fig 8. Precision Formula

- *Recall*
  Calculate by dividing the correct data by the total number of true data (True Positive) and incorrect data (False Negative). Here is the recall equation:

$$Recall = \frac{TP}{TP+FN} \times 100\%$$

Fig 9. Recall Formula

III. Methodology

A. SEMMA

SEMMA refers to stages or methodologies that can be used for both data mining and machine learning projects. This framework has been developed by the Suite of Analytics (SAS) Institute and has proven to be widely used by the industry to date. SEMMA stands for the methodological process which consists of five stages, namely Sample (sample), Explore (exploration), Modify (modification), Model (modeling), and Asses (assessment) [19]. The following is a more complete explanation regarding the five stages of SEMMA [20]:

- Sample - This stage is the stage where sampling or selection of representative data from the data subset will be carried out. Sampling or selection will be taken from a larger dataset with the aim of finding the most influential variables
- Explore – Furthermore, after sampling, exploration will be carried out with the aim of further understanding the correlation of each variable that exists. In this stage, data visualization will generally be carried out which is useful for studying the results of previous explorations, and there will also be descriptive statistics calculations that can be performed.
- Modify - After the exploration stage, it will enter the modification stage where the data will be cleaned or changed according to the goals that have been determined. This is done so that the data is ready to be used when doing modeling.
- Model - With the completion of the modification process, the data has been cleaned and transformed so it will enter the modeling stage. The modeling stage is the stage where machine learning processes will be implemented to produce the expected output.
- Assess - In the last stage, the model has been created and an evaluation will be carried out on the model to find out how effective and useful the model that has been made is for the stated goals. The performance of the model is usually the benchmark at this stage.

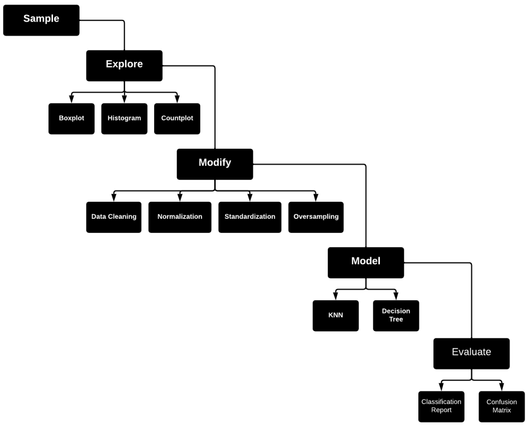


Fig 10. Illustration of Research Methodology

IV. Results and Discussion

A. SEMMA

- *Sample*

The data in this study were taken from various sources such as the results of the data mining process from the official IDX website on Github and also Kaggle. Then, the data that has been retrieved is combined using python. There is a main dataset named List of Shares.csv, where this dataset has the attributes Code, Name, ListingDate, Shares, ListingBoard, Sector, LastPrice, MarketCap, MinutesFirstAdded, MinutesLastUpdated, HourlyFirstAdded, HourlyLastUpdated, DailyFirstAdded, dan DailyLastUpdated. Then, there is a data mining results folder which contains stock details of the stocks on the official IDX website. However, because the data mining method used is not perfect, there are several stocks that do not contain details. From there, the author checks, then combines the main dataset with the detailed data, where the final dataset produced is as follows:

No.	Attribute	Description
1.	Code	Company stock code name.
2.	Name	Company name.
3.	ListingDate	The date the company's stock first went public.
4.	Shares	The number of outstanding shares and the percentage of shares owned by public investors.
5.	ListingBoard	The classes are divided into 3 categories.
6.	MarketCap	The company's market capitalization is calculated by calculating multiply the share price by the number of shares outstanding.
7.	DailyLastUpdate	The last date the company's shares were updated.
8.	Operational Lifetime	The length of time the company has been listed on IDX.
9.	Avg. Value	Average daily trading of company stock.
10.	Avg. Frequency	The average trading activity of the company's stock within a certain period of time.

Table 1. Information Data

Not only that, feature engineering has been carried out for the Avg column. Value, Avg Frequency, and Operational Lifetime. Where is the avg column. values and avg. frequency is obtained from the process of calculating the average of each detailed dataset of the stock. In addition, the operational lifetime column is obtained from the results of reducing the DailyLastUpdated and DailyFirstAdded columns.

- *Explore*

Furthermore, in the explore stage, exploratory data analysis will be carried out where the author will find out more about the properties of the dataset that is owned. At this stage, several things that are seen and learned are outliers in the data, data distribution, and finally the amount of data in each class to see the balance level of the data.

First, we will look at the outliers in each column in the dataset, for this a boxplot visualization will be carried out. The following is a visualization that has been done:

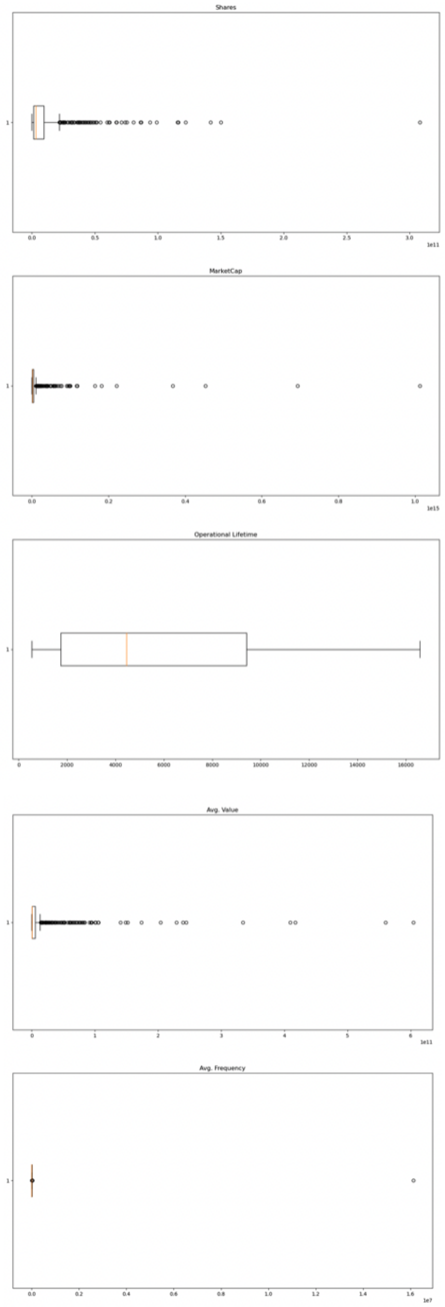


Fig 11. Boxplot Visualization

It can be seen that there are extreme outliers in several columns such as Shares, MarketCap and Avg. Value. From here the author knows that normalization must be carried out at a later stage.

After looking at the outliers, then we will look at the distribution of the data. To do this, a Histogram visualization will be performed using the previous five columns. The following is the result of the Histogram visualization that was carried out:

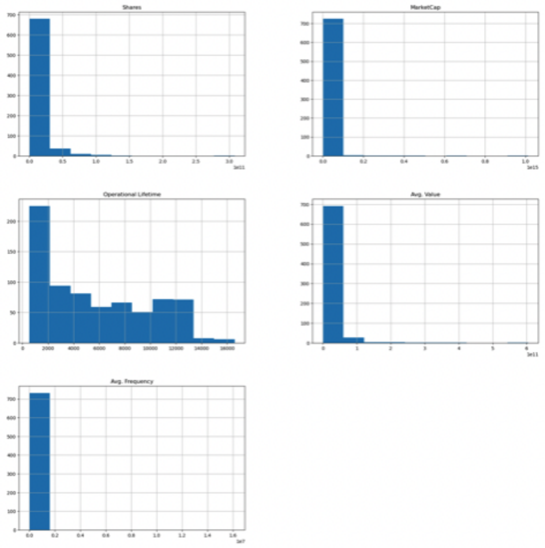


Fig 12. Histogram Visualization

From the Histogram visualization, it can be seen that the data distribution does not follow the normal or Gaussian distribution. So from this, after normalization will be standardized at a later stage. Final. Countplot visualization will be carried out to see the amount of data for each class in the dataset. The following is the result of the Countplot visualization that was carried out:

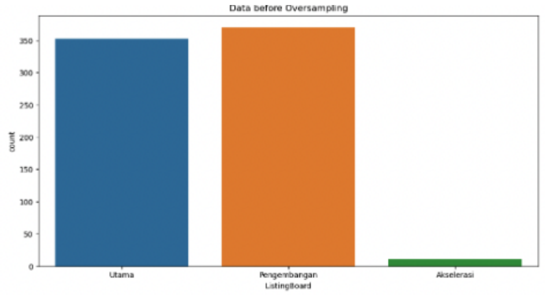
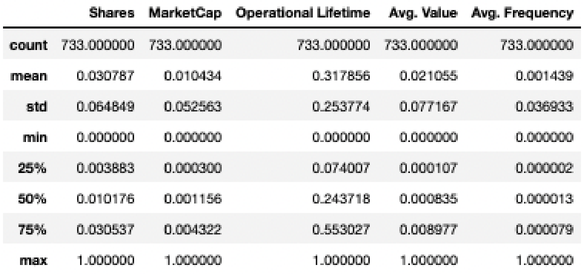


Fig 13. Countplot Visualization

Visualization of the Countplot above, it can be seen that the data for the 'Acceleration' class is very small which means that the data is very imbalanced. Therefore, hybrid oversampling will be carried out at a later stage.

- *Modify*

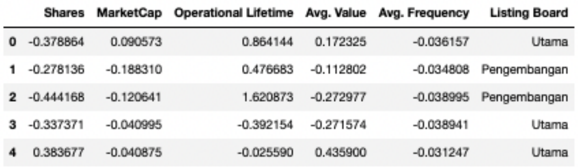
Furthermore, in the modify stage, modifications will be made to the dataset in accordance with the exploration results in the previous stage. First of all it will be normalized using MinMaxScaler, as is known from the previous stage, normalization is carried out in this study to ensure the scale values of the dataset attributes are not too far away. Normalization itself will make the attributes in the dataset have a minimum value of 0 and a maximum of 1. The following are the basic statistics of the dataset after normalization:



	Shares	MarketCap	Operational Lifetime	Avg. Value	Avg. Frequency
count	733.000000	733.000000	733.000000	733.000000	733.000000
mean	0.030787	0.010434	0.317856	0.021055	0.001439
std	0.064849	0.052563	0.253774	0.077167	0.036933
min	0.000000	0.000000	0.000000	0.000000	0.000000
25%	0.003883	0.000300	0.074007	0.000107	0.000002
50%	0.010176	0.001156	0.243718	0.000835	0.000013
75%	0.030537	0.004322	0.553027	0.008977	0.000079
max	1.000000	1.000000	1.000000	1.000000	1.000000

Fig 14. Basic Statistics

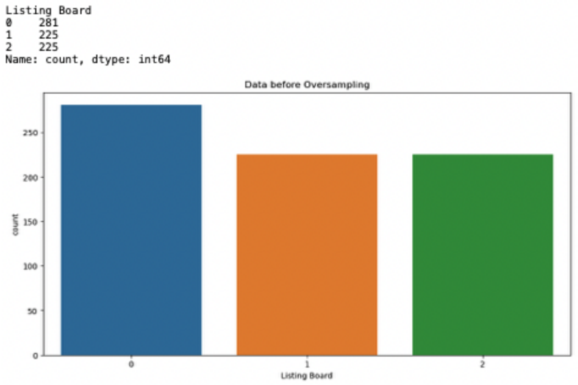
Furthermore, after normalization, standardization will be carried out using StandardScaler. The purpose of standardizing in this particular research is to ensure that the data is further transformed to have a mean of zero and standard deviation of one. Doing standardization here also helps the model to eliminate any potential bias that may come from the dataset. The following is the result of the dataset after standardization using StandardScaler:



	Shares	MarketCap	Operational Lifetime	Avg. Value	Avg. Frequency	Listing Board
0	-0.378864	0.090573	0.864144	0.172325	-0.036157	Utama
1	-0.278136	-0.188310	0.476683	-0.112802	-0.034808	Pengembangan
2	-0.444168	-0.120641	1.620873	-0.272977	-0.038995	Pengembangan
3	-0.337371	-0.040995	-0.392154	-0.271574	-0.038941	Utama
4	0.383677	-0.040875	-0.025590	0.435900	-0.031247	Utama

Fig 15. Data AfterStandardScaler

After standardization, a class column will be mapped to change from categorical to numerical. Then after that, outlier removal will be carried out using the z-score technique. Where before removing the outliers there were 733 data and after removing the outliers using the z-score there were 711 data. Finally, a hybrid oversampling technique will be carried out using SMOTETomek. The hybrid oversampling technique combines both oversampling and undersampling. Where in this study it is used to make the dataset more balanced. The following is the result of the dataset after hybrid oversampling and its visualization:

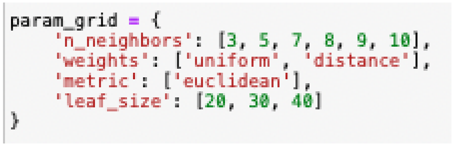


```
Listing Board
0    281
1    225
2    225
Name: count, dtype: int64
```

Fig 16. Countplot Results and Visualization

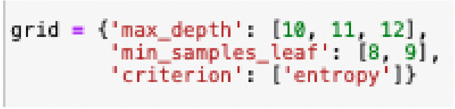
- *Model*

Furthermore, at the modeling stage, modeling will be carried out according to the algorithm that has been selected, namely K-Nearest Neighbors and Decision Tree. The modified data will be divided into 2 sets, namely the training and testing sets. Then each set will have X and Y where X is the independent variable and Y is the dependent variable. To determine the parameters to be used, the Grid Search library has been used to find the best combination of parameters for the model according to the data used. For the KNN and Decision Tree models, the parameter combinations to be checked using GridSearch are as follows:



```
param_grid = {
    'n_neighbors': [3, 5, 7, 8, 9, 10],
    'weights': ['uniform', 'distance'],
    'metric': ['euclidean'],
    'leaf_size': [20, 30, 40]
}
```

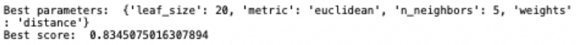
Fig 17. GridSearch KNN



```
grid = {'max_depth': [10, 11, 12],
        'min_samples_leaf': [8, 9],
        'criterion': ['entropy']}
```

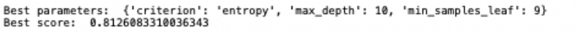
Fig 18. GridSearch Decision Tree

Based on the predefined grid, here are the results returned by GridSearch and their values:



```
Best parameters:  {'leaf_size': 20, 'metric': 'euclidean', 'n_neighbors': 5, 'weights'
: 'distance'}
Best score:  0.8345075016307894
```

Fig 19. Best Parameter + Score KNN



```
Best parameters:  {'criterion': 'entropy', 'max_depth': 10, 'min_samples_leaf': 9}
Best score:  0.8126083310036343
```

Fig 20. Best Parameter + Score Decision Tree

- *Assess*

Finally, at the assess stage, the evaluation method that will be used is the classification report, where the author will look at precision, recall, f1-score, and support. After viewing the classification report of each model, a confusion matrix visualization will then be performed to see the True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN) values. Finally, a visualization of the decision tree model that has been made along with the feature importance of each attribute will be visualized, and also for KNN the class distribution will be seen using the Scatterplot visualization.

This study uses the SEMMA research method as previously described. Therefore, the following are the results and discussion of this study according to the steps of the SEMMA method.

A. K-Nearest Neighbors

	precision	recall	f1-score	support
0	0.40	1.00	0.57	2
1	0.81	0.72	0.76	85
2	0.67	0.75	0.71	56
accuracy			0.73	143
macro avg	0.63	0.82	0.68	143
weighted avg	0.75	0.73	0.74	143

Table 2. Evaluation K-Nearest Neighbors

It can be seen in table 2 above, the results of the classification process using the KNN algorithm show the values of precision, recall, F1-score and support for each class. In class 0, the value of precision is 0.40, recall is 1.00, F1-score is 0.57 and support is 2.

Then in class 1, the precision value is 0.81, recall is 0.72, F1-score is 0.76 and support is 85. Finally in class 2, precision value is 0.67, recall is 0.75, F1-score is 0.71 and support is 56. Accuracy value obtained from the KNN algorithm model is 0.73

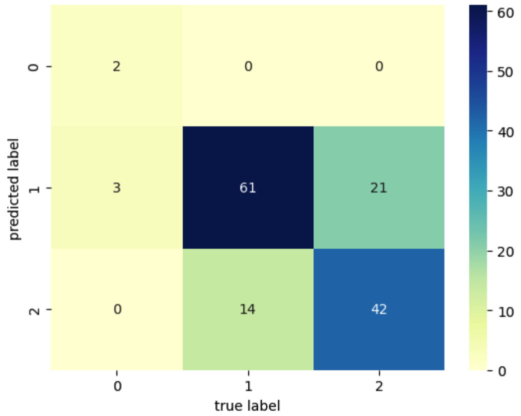


Fig 21. Confusion Matrix KNN

The picture above is a heatmap visualization of the confusion matrix for the KNN algorithm, you can see the relationship between true labels and predicted labels. For class 0, there are 2 data which are class 0 and are predicted to enter class 0 (True Positive) and there are 0 data respectively with classes 1 and 2 which are predicted to enter class 0 (False Positive), then there are 3 and The 0 predicted data fall into class 1 and 2 which is actually class 0 (False Negative). Finally, there are 61, 21, 14, and 42 true negative data for class 0.

For class 1, there are 61 data which are class 1 and are predicted to enter class 1 (True Positive) and there are 3 and 21 data with class 0 and 2 which are predicted to enter class 1 (False Positive), then there are as many as 0 and 14 the predicted data fall into class 0 and 2 where in fact it is class 1 (False Negative). Finally, there are 2, 0, 10, and 42 true negative data for class 1.

For class 2, there are 42 data which are class 2 and predicted to enter class 2 (True Positive) and there are 0 and 14 data with class 0 and 1 which are predicted to enter class 2 (False Positive), then there are 0 and 21 the predicted data fall into class 0 and 1 where it is actually class 2 (False Negative). Finally, there are 2, 0, 3, and 61 true negative data for class 2.

B. Decision Tree Classifier

	precision	recall	f1-score	support
0	1.00	1.00	1.00	2
1	0.80	0.80	0.80	85
2	0.70	0.70	0.70	56
accuracy			0.76	143

macro avg	0.83	0.83	0.83	143
weighted avg	0.76	0.76	0.76	143

Table 3. Evaluation Decision Tree

It can be seen in table 3 above, the results of the classification process using the Decision Tree algorithm show the recall precision value, F1-score for each class. In class 0, precision is 1.00, recall is 1.00, F1-score is 1.00 and support is 2.

Then in class 1, the precision value is 0.80, recall is 0.80, F1-score is 0.80 and support is 85. Finally in class 2, precision value is 0.70, recall is 0.70, F1-score is 0.70 and support is 56. Accuracy value obtained from the Decision Tree algorithm model is 0.76.

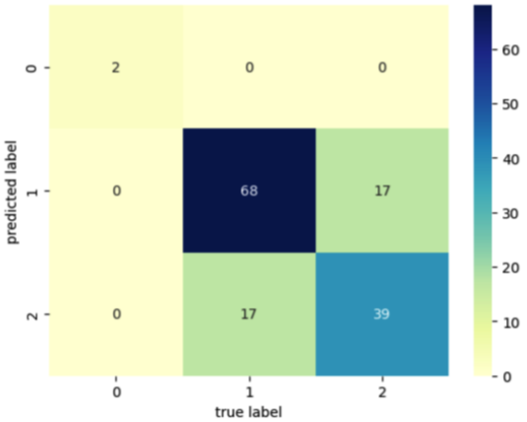


Fig 22. Confusion Matrix Decision Tree

The image above is a heatmap visualization of the confusion matrix for the Decision Tree algorithm, we can see the relationship between true labels and predicted labels. For class 0, there are 2 data which are class 0 and are predicted to enter class 0 (True Positive) and there are 0 data respectively with classes 1 and 2 which are predicted to enter class 0 (False Positive), then there are as many as 0 and The 0 predicted data fall into class 1 and 2 which is actually class 0 (False Negative). Finally there are 68, 17, 17, and 39 true negative data for class 0.

For class 1, there are 68 data which are class 1 and are predicted to enter class 1 (True Positive) and there are 0 and 17 data with class 0 and 2 which are predicted to enter class 1 (False Positive), then there are as many as 0 and 17 the predicted data fall into class 0 and 2 where in fact it is class 1 (False Negative). Finally, there are 2, 0, 0, and 39 true negative data for class 1.

For class 2, there are 39 data which are class 2 and are predicted to enter class 2 (True Positive) and there are 0 and 17 data with class 0 and 1 which are predicted to enter class 2 (False Positive), then there are as many as 0 and 17 the predicted data fall into class 0 and 1 where it is actually class 2 (False Negative). Finally there are 2, 0, 0, and 68 true negative data for class 2.

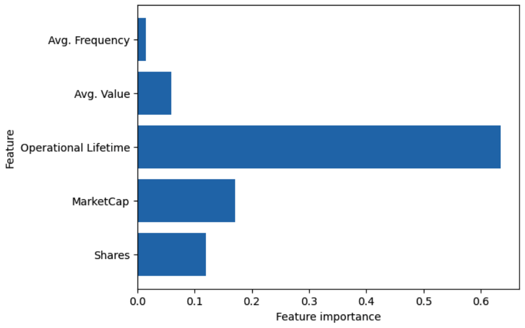


Fig 23. Feature Important Decision Tree

The visualization above is carried out to see the importance of the features of the Decision Tree, here is Avg. Frequency, Avg. Value, Operational Lifetime, MarketCap and Shares. This is done to see which features have the most influence on the Decision Tree model, where we can see that Operational Lifetime has the largest number, namely 0.6.

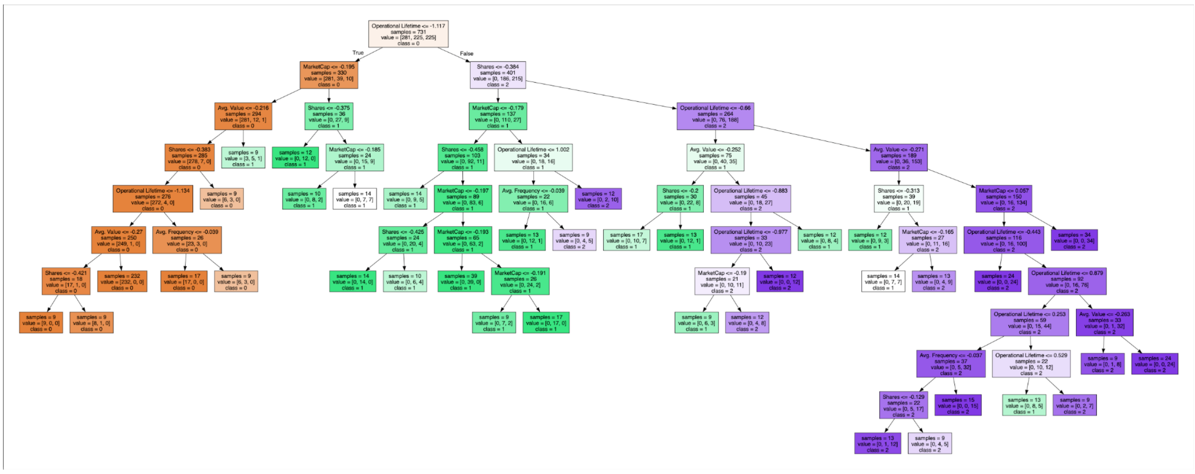

Operational Lifetime <= -1.117 samples = 731 value = [281, 225, 225] class = 0
True
False
MarketCap <= -0.195 samples = 330 value = [281, 39, 10] class = 0
Shares <= -0.384 samples = 401 value = [0, 186, 215] class = 2
Avg. Value <= -0.216 samples = 294 value = [281, 12, 1] class = 0
Shares <= -0.375 samples = 36 value = [0, 27, 9] class = 1
MarketCap <= -0.179 samples = 137 value = [0, 110, 27] class = 1
Operational Lifetime <= -0.66 samples = 264 value = [0, 76, 188] class = 2
Shares <= -0.383 samples = 285 value = [278, 7, 0] class = 0
samples = 9 value = [3, 5, 1] class = 1
samples = 12 value = [0, 12, 0] class = 1
MarketCap <= -0.185 samples = 24 value = [0, 15, 9] class = 1
Shares <= -0.458 samples = 103 value = [0, 92, 11] class = 1
Operational Lifetime <= 1.002 samples = 34 value = [0, 18, 16] class = 1
Avg. Value <= -0.252 samples = 75 value = [0, 40, 35] class = 1
Avg. Value <= -0.271 samples = 189 value = [0, 36, 153] class = 2
Operational Lifetime <= -1.134 samples = 276 value = [272, 4, 0] class = 0
samples = 9 value = [6, 3, 0] class = 0
samples = 10 value = [0, 8, 2] class = 1
samples = 14 value = [0, 7, 7] class = 1
samples = 14 value = [0, 9, 5] class = 1
MarketCap <= -0.197 samples = 89 value = [0, 83, 6] class = 1
Avg. Frequency <= -0.039 samples = 22 value = [0, 16, 6] class = 1
samples = 12 value = [0, 2, 10] class = 2
Shares <= -0.2 samples = 30 value = [0, 22, 8] class = 1
Operational Lifetime <= -0.883 samples = 45 value = [0, 18, 27] class = 2
Shares <= -0.313 samples = 39 value = [0, 20, 19] class = 1
MarketCap <= 0.057 samples = 150 value = [0, 16, 134] class = 2
Avg. Value <= -0.27 samples = 250 value = [249, 1, 0] class = 0
Avg. Frequency <= -0.039 samples = 26 value = [23, 3, 0] class = 0
Shares <= -0.425 samples = 24 value = [0, 20, 4] class = 1
MarketCap <= -0.193 samples = 65 value = [0, 63, 2] class = 1
samples = 13 value = [0, 12, 1] class = 1
samples = 9 value = [0, 4, 5] class = 2
samples = 17 value = [0, 10, 7] class = 1
samples = 13 value = [0, 12, 1] class = 1
Operational Lifetime <= -0.977 samples = 33 value = [0, 10, 23] class = 2
samples = 12 value = [0, 8, 4] class = 1
samples = 12 value = [0, 9, 3] class = 1
MarketCap <= -0.165 samples = 27 value = [0, 11, 16] class = 2
Operational Lifetime <= -0.443 samples = 116 value = [0, 16, 100] class = 2
samples = 34 value = [0, 0, 34] class = 2
Shares <= -0.421 samples = 18 value = [17, 1, 0] class = 0
samples = 232 value = [232, 0, 0] class = 0
samples = 17 value = [17, 0, 0] class = 0
samples = 9 value = [6, 3, 0] class = 0
samples = 14 value = [0, 14, 0] class = 1
samples = 10 value = [0, 6, 4] class = 1
samples = 39 value = [0, 39, 0] class = 1
MarketCap <= -0.191 samples = 26 value = [0, 24, 2] class = 1
MarketCap <= -0.19 samples = 21 value = [0, 10, 11] class = 2
samples = 12 value = [0, 0, 12] class = 2
samples = 14 value = [0, 7, 7] class = 1
samples = 13 value = [0, 4, 9] class = 2
samples = 24 value = [0, 0, 24] class = 2
Operational Lifetime <= 0.879 samples = 92 value = [0, 16, 76] class = 2
samples = 9 value = [9, 0, 0] class = 0
samples = 9 value = [8, 1, 0] class = 0
samples = 9 value = [0, 7, 2] class = 1
samples = 17 value = [0, 17, 0] class = 1
samples = 9 value = [0, 6, 3] class = 1
samples = 12 value = [0, 4, 8] class = 2
Operational Lifetime <= 0.253 samples = 59 value = [0, 15, 44] class = 2
Avg. Value <= -0.263 samples = 33 value = [0, 1, 32] class = 2
Avg. Frequency <= -0.037 samples = 37 value = [0, 5, 32] class = 2
Operational Lifetime <= 0.529 samples = 22 value = [0, 10, 12] class = 2
samples = 9 value = [0, 1, 8] class = 2
samples = 24 value = [0, 0, 24] class = 2
Shares <= -0.129 samples = 22 value = [0, 5, 17] class = 2
samples = 15 value = [0, 0, 15] class = 2
samples = 13 value = [0, 8, 5] class = 1
samples = 9 value = [0, 2, 7] class = 2
samples = 13 value = [0, 1, 12] class = 2
samples = 9 value = [0, 4, 5] class = 2

Fig 24. Visualization Decision Tree

The image above shows the results of the Decision Tree modeling, where we can see the labels of the sample data based on the features they have. The final result of this Decision Tree that has a root node is Operational Lifetime.

V. Conclusion

As a conclusion from this research, from the two models that have been made and the results and performance compared, the following are some points that can be taken to answer the formulated problems:

(1)The most influential factor in classifying shares for the three available listing boards is Operational Lifetime; (2)The factors used in this study are less able to classify shares to the listing board; (3)The algorithm that has better results and performance is the Decision Tree algorithm with an accuracy of 76%.

Decision Tree can be an algorithm with better performance and results because Decision Tree is an algorithm that can handle datasets that have more complex correlations between features. On the other hand, KNN requires 'similar' data. That is the reason why the Decision Tree algorithm gets 76% accuracy and KNN gets 73% accuracy

References

[1] D. Shah, H. Isah, and F. Zulkernine, "Stock Market Analysis: A Review and Taxonomy of Prediction Techniques," vol. 7(2), 2019.

[2] K. Alkhatib, H. Najadat, I. Hmeidi, and M. K. A. Shatnawi, "Stock price prediction using k-nearest neighbor (knn) algorithm," International Journal of Business, Humanities and Technology, vol. 3, no. 3, pp. 32–44, 2013

[3] Wiyono and T. Abidin, "Comparative study of machine learning knn, svm, and decision tree algorithm to predict student's performance," International Journal of Research-Granthaalayah, vol. 7, no. 1, pp. 190–196, 2019.

[4] D. Sree and S. Bindu, Data Analytics: Why Data Normalization (2018), https://www.academia.edu/download/74142822/9582.pdf .

[5] V. R. Prasetyo, M. Mercifia, A. Averina, L. Sunyoto, and Budiarjo, "Jurnal Ilmiah Nero Vol. 7 no. 1 - repository.ubaya.ac.id," PREDIKSI RATING FILM PADA WEBSITE IMDB MENGGUNAKAN METODE NEURAL NETWORK (2022), http://repository.ubaya.ac.id/41805/1/268-890-1-PB.pdf.

[6] X. Gong, F. Zhang, T. Lu, and W. You, "Comparative analysis of three outlier detection methods in univariate data sets," 2022 3rd International Conference on Electronic Communication and Artificial Intelligence (IWECAI), pp. 209–213, 2022. doi:10.1109/iwecai55315.2022.00048

[7] M. Khushi et al., "A comparative performance analysis of data resampling methods on Imbalance Medical Data," IEEE Access, vol. 9, pp. 109960–109975, 2021. doi:10.1109/access.2021.3102399

[8] Z. Wang, C. Wu, K. Zheng, X. Niu, and X. Wang, "Smotetomek-based resampling for personality recognition," IEEE Access, vol. 7, pp. 129678–129689, 2019. doi:10.1109/access.2019.2940061

[9] E. Etriyanti, "Perbandingan tingkat akurasi metode knn dan decision tree dalam memprediksi

lama studi mahasiswa," Jurnal Ilmiah Binary STMIK Bina Nusantara Jaya, vol. 3, no. 1, 2021.

[10] P. R. Sihombing and A. M. Arsani, "Comparison of Machine Learning Methods in Classifying Poverty in Indonesia in 2018," Jurnal Teknik Informatika, vol. 2(1), pp. 51–56, 6 2021.

[11] E. R. Tauran, "Prediksi harga saham pt bank central asia tbk berdasarkan data dari bursa efek indonesia menggunakan metode k-nearest neighbors (knn),"TeIKa, vol. 11, no. 2, pp. 123–129, 2021.

[12] S. H. S. Robianto and U. Ristian, "PENERAPAN METODE DECISION TREE UNTUK MENGKLASIFIKASIKAN MUTU BUAH JERUK BERDASARKAN FITUR WARNA DAN UKURAN," Jurnal Komputer dan Aplikasi, vol. 9(1), pp. 76–86, 2021.

[13] M. Nabipour, P. Nayyeri, H. Jabani, S. Shahab, and A. Mosavi, "Predicting stock market trends using machine learning and deep learning algorithms via continuous and binary data; a comparative analysis," IEEE Access, vol. 8, pp. 150 199–150 212, 2020.

[14] A. Ardiyansyah, P. A. Rahayuningsih, and R. Maulana, "Analisis perbandingan algoritma klasifikasi data mining untuk dataset blogger dengan rapid miner," Jurnal Khatulistiwa Informatika, vol. 6, no. 1, 6 2018.

[15] A. T. Saragih, D. Afrida, and R. F. Dinara, "Klasifikasi jumlah kendaraan di sumatera utara dan sumatera barat menggunakan algoritma naive bayes," JCom (Journal of Computer), vol. 3, no. 1, pp. 11–16, 2023.

[16] R. Ainun, "Penilaian kelayakan pinjaman nasabah menggunakan logika fuzzy metode mamdani," Ph.D. dissertation, Institut Teknologi Kalimantan, 2021.

[17] R. Rindiyani, A. Primadewi, M. Maimunah, and A. H. Purwantini, "Klasifikasi penjualan berdasarkan platform pada umkm omah branded menggunakan random forest," JURIKOM (Jurnal Riset Komputer), vol. 9, no. 5, pp.,1520–1529, 2022.

[18] Y. Suryana and T. W. Sen, "The prediction of gold price movement by comparing naive bayes, support vector machine, and k-nn," Jurnal Informatika dan Sains, vol. 4, no. 2, pp. 112–120, 2021.

[19] Firas, Omari. "A combination of SEMMA & CRISP-DM models for effectively handling big data using formal concept analysis based knowledge discovery: A data mining approach." World Journal of Advanced Engineering Technology and Sciences, vol. 8, no. 1, 2023.

[20] N. Hotz, "What is SEMMA?," Data Science Process Alliance, https://www.datascience-pm.com/semma/, 2023