Journal of Science Technology (JoSTec)
e-ISSN. 2830-7275
Published by: Inovasi Pratama Internasional. Ltd
Journal homepage: https://ejournal.ipinternasional.com/index.php/jostec
DOI: https://doi.org/10.55299/jostec.v4i1.251

KNN and XGBoost Algorithms for Lung Cancer Prediction

M. Rhifky Wayahdi[1], Fahmi Ruziq[2]
[1,2]Department of Information System, Faculty of Technology, Battuta University, Indonesia
[1]muhammadrhifkywayahdi@gmail.com, [2]fahmiruziq89@gmail.com

ARTICLE INFO

Article history:

Received December 10, 2022
Revised December 12, 2022
Accepted December 17, 2022

Keywords:

K-nearest neighbor algorithm
XGBoost algorithm
Machine learning
Classification algorithms
Cancer prediction

ABSTRACT

In this paper, the K-Nearest Neighbor and XGBoost algorithms will be implemented in lung cancer prediction. This prediction is important because lung cancer is one of the highest causes of death worldwide. Prediction and diagnosis of this cancer can be done with machine learning algorithms such as K-Nearest Neighbor and XGBoost. Based on the results of the analysis and testing of the K-Nearest Neighbor algorithm and the XGBoost algorithm, the results show that the two algorithms obtain a very good level of accuracy, as well as obtain a balanced precision, recall, and f1-score. But in this case the XGBoost algorithm tends to be better than the KNN algorithm in recognizing a given data pattern.

Corresponding Author:

M. Rhifky Wayahdi,
Department of Information System,
Faculty of Technology, Battuta University, Medan, Indonesia.
Email: muhammadrhifkywayahdi@gmail.com

1. INTRODUCTION

Cancer is one of the causes of death in the world which occurs due to cells in the body that reproduce abnormally (American Cancer Society, 2017). One type of cancer is lung cancer. Lung cancer is the leading cause of death worldwide and has the highest morbidity among all types of cancer. It is estimated that around 228,820 new cases of lung cancer will be diagnosed in the United States (US) in 2020 and up to 135,720 patients will die from the disease (Yuan, et al., 2022).

Cancer prediction and diagnosis is a complex subject of interest to researchers worldwide because of the high morbidity and mortality of the disease (Sharma & Rani, 2021). Early diagnosis and prognosis of a type of cancer has become a necessity in cancer research because it can help clinical therapy of patients. Early detection of cancer increases the chances of successful therapy (Gu, et al., 2022). Early diagnosis of cancer by developing efficient computational prediction models can play an important role in this context (Nardini, 2020).

In the last few decades, diagnosing various diseases has been carried out using machine learning methodologies (Wu, et al., 2018). In its implementation, machine learning has many methods that can be used to handle classification, clustering, and other data management. Classification is a data mining method that can help make predictions so that it can estimate the class of an object whose label is unknown.

K-Nearest Neighbor (KNN) is a method that is most widely used in classification (Wayahdi, et al., 2020), pattern recognition, and text categories (Sinaga, et al., 2020). In a study on Indoor Localization using the K-Nearest Neighbor (KNN) and Backpropagation methods, it was found that the KNN method produces better accuracy than the Backpropagation method (Adege, et al., 2018). And in the study of (Jaafar, et al.,

2016) using the KNN method, to classify and optimize hand-based biometric image databases, to get a better percentage.

Another classification method is the eXtreme Gradient Boosting (XGBoost) method. In several studies using this method can produce very good accuracy. In a study by (Cherif & Kortebi, 2019) which addresses the problem of traffic classification using the XGBoost algorithm. The performance evaluation results obtained an accuracy of 99.5%, which is the most accurate compared to other algorithms. Based on these problems, the authors are interested in applying the KNN and XGBoost classification methods in this study.

2. RESEARCH METHOD

2.1. K-Nearest Neighbor (KNN)

K-Nearest Neighbor (KNN) is the most suitable classical classification method (Pan, et al., 2017) for its simplicity, adaptability, and performance. The KNN method can be used for classification in pattern recognition, but can also be used for prediction or regression (Lei & Zhaowei, 2017). The KNN method calculation algorithm is as follows:

a. Set the number of k (number of nearest neighbors to be selected).
b. Calculate the distance between the data to be classified and all training data using the Euclidean distance with the following equation:

$$D_i = \sqrt{\sum_{i,j=1}^{n} (X - X_{i,j})^2}$$

c. Sort ascending distance that has been formed.
d. Determine the shortest distance with a predetermined k.
e. Assemble the appropriate class.
f. Find the number of classes from the most neighbors and set that class as the data class to be evaluated next.

2.2. Extreme Gradient Boosting (XGBoost)

XGBoost is one of the well-working algorithms used for supervised learning. This algorithm can be used for regression and classification problems. XGBoost is loved by data scientists because of its fast execution (Osman, et al., 2021).

XGBoost is basically an ensemble method based on gradient amplified trees. The prediction result is the sum of the scores predicted by K trees, as shown in the formula below:

$$\hat{y}_i = \sum_{k=1}^{k} f_k(x_i), f_k \in F$$

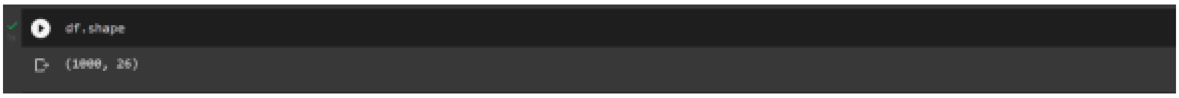
Where x_i is the ith training sample, $f_k(x_i)$ is the score for the kth tree, and *F* is the function space containing all gradient-enhanced trees. The objective function can be optimized by the following formula (Zhang, et al., 2021):

$$\text{obj}\ (\theta) = \sum_{i=1}^{n} l_{(y_i, \hat{y}_i)} + \sum_{k=1}^{K} \Omega(f_k)$$

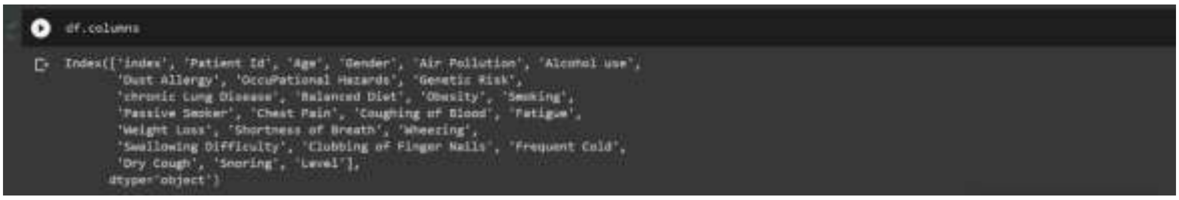
3. RESULTS AND DISCUSSIONS

3.1. Data Collection

The data used in this study is lung cancer data. This data contains 26 columns and 1000 rows, where 25 features columns (attributes/variables) and 1 label column (target).



```
df.shape

(1000, 26)
```



```
df.columns

Index(['index', 'Patient Id', 'Age', 'Gender', 'Air Pollution', 'Alcohol use',
       'Dust Allergy', 'OccuPational Hazards', 'Genetic Risk',
       'chronic Lung Disease', 'Balanced Diet', 'Obesity', 'Smoking',
       'Passive Smoker', 'Chest Pain', 'Coughing of Blood', 'Fatigue',
       'Weight Loss', 'Shortness of Breath', 'Wheezing',
       'Swallowing Difficulty', 'Clubbing of Finger Nails', 'Frequent Cold',
       'Dry Cough', 'Snoring', 'Level'],
      dtype='object')
```

3.2. Data Cleaning

Data cleaning is a process for detecting and correcting errors in data so that the data becomes clean and of good quality. The data cleaning process has been carried out by filling in blank data, unreasonable values, and duplicate data. The data can be ensured that it is valid, complete, consistent, and uniform.

3.3. Exploratory Data Analysis (EDA)

Exploratory data analysis is done by looking at the correlation between two features (variables/attributes). Figure 1 shows the correlation heatmap between features.

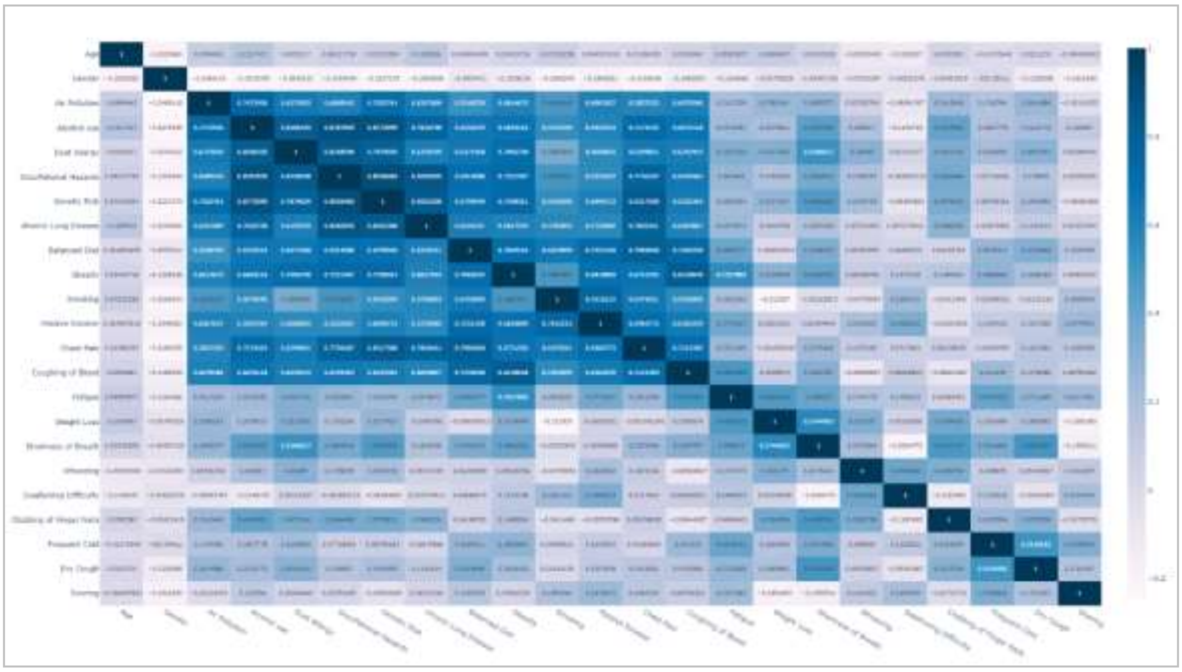


Figure 1. Correlation heatmap between features

In Figure 1 it can be seen that the highest correlation is in the occupational hazards and genetic risk features with a correlation value of 0.8930485, alcohol use and occupational hazards with a correlation value of 0.8787859, and alcohol use and genetic risk with a correlation value of 0.8772099.

There are 3 parameters to read the correlation value, namely:

-1: Full negative correlation. This means that if the variable goes up, the other variables go down and are fully correlated.

0: No correlation at all. This means that the two variables are not dependent at all. If one goes up, you can't predict with any probability what will happen to the other.

1: Full correlation. This means that if one of the variables increases, so does the other.

Next will be analyzed the correlation between features and labels (targets). Figure 2 shows the results of the chi-square test between features and labels.

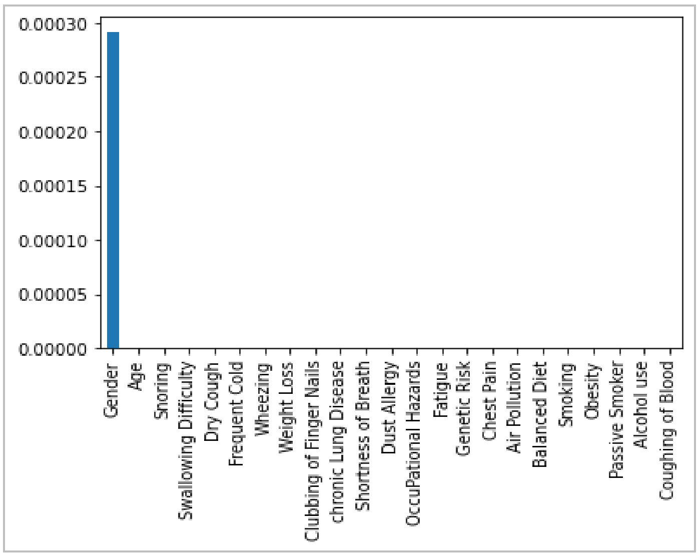
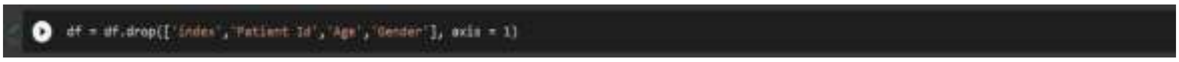


Figure 2. Chi-square test analysis

Based on the chi-square test analysis, all features tend to be related to the label because there is no p-value greater than 0.05 (alpha value).

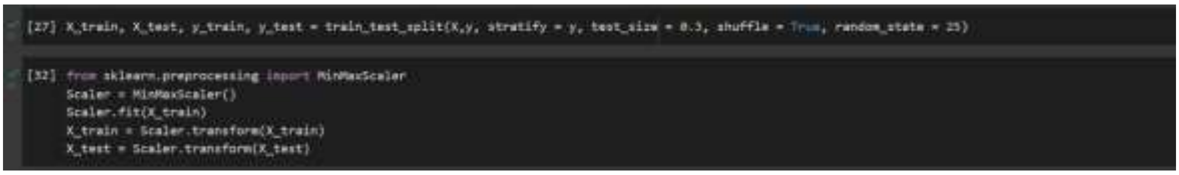
3.4. Data Pre-processing

Based on the context of the data and the purpose of the research, there are irrelevant columns, namely index, patient id, age, and gender.



```
df = df.drop(['index','Patient Id','Age','Gender'], axis = 1)
```

Then divide (split) the data into training data and test data with a ratio of 70:30. Then normalize the data or rescaling the numeric data to a range of 0 to 1 with a min-max scaler. This is done if there is numeric data with large values or a large range of data compared to other numeric data, with the aim of equalizing the scale with other data (not too different) so that machine learning does not tend to be biased towards the influence of large numeric data.



```
[27] X_train, X_test, y_train, y_test = train_test_split(X,y, stratify = y, test_size = 0.3, shuffle = True, random_state = 25)

[32] from sklearn.preprocessing import MinMaxScaler
     Scaler = MinMaxScaler()
     Scaler.fit(X_train)
     X_train = Scaler.transform(X_train)
     X_test = Scaler.transform(X_test)
```

3.5. Model Building

3.5.1. KNN Algorithm

The first model building stage uses the K-Nearest Neighbor (KNN) algorithm where this algorithm is a simple and popular algorithm. This algorithm requires a value of k for grouping data, similar to using the k-means method (Wayahdi, et al., 2020). Determination of the value of k based on MSE can be seen in Figure 3.

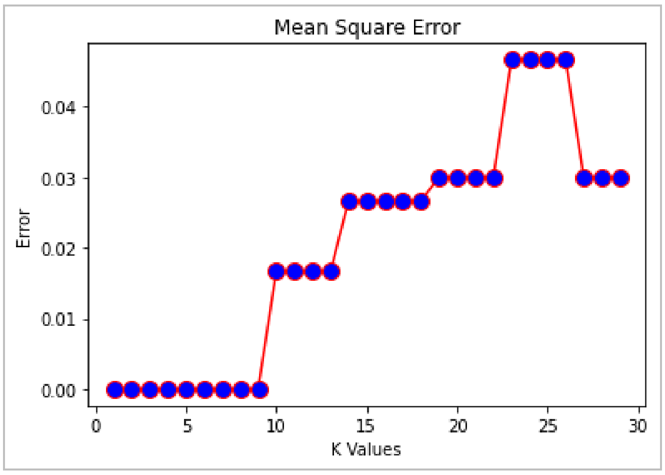
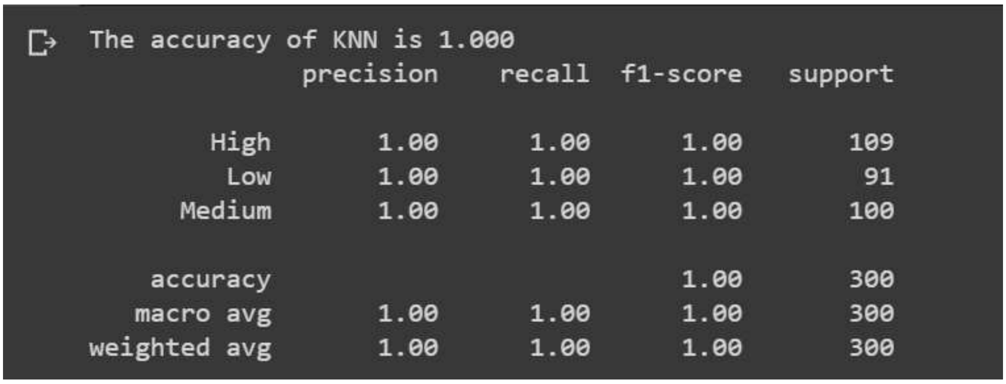


Figure 3. MSE at k values

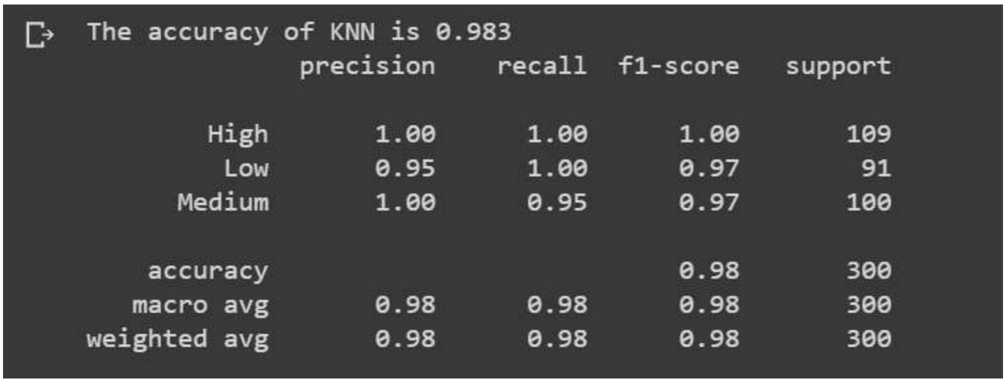
In Figure 3 it can be seen that the best k values are in the range 1 to 9 with an MSE of 0%, but overall from a k value of 10 onwards it is still quite good. Figure 4 shows the results of the accuracy, precision, recall, and f1-score modeling results from KNN with k value = 3.



```
The accuracy of KNN is 1.000
              precision    recall  f1-score   support

        High       1.00      1.00      1.00       109
         Low       1.00      1.00      1.00        91
      Medium       1.00      1.00      1.00       100

    accuracy                           1.00       300
   macro avg       1.00      1.00      1.00       300
weighted avg       1.00      1.00      1.00       300
```

Figure 4. Modeling result with KNN (k = 3)

In Figure 4 it can be seen that the accuracy of the KNN model is 100% and the precision, recall, and f1-score values are also 100%, this shows that the model built with the KNN algorithm is good and can recognize the given pattern. Next, a trial will be carried out with k value = 11, the results are shown in Figure 5.



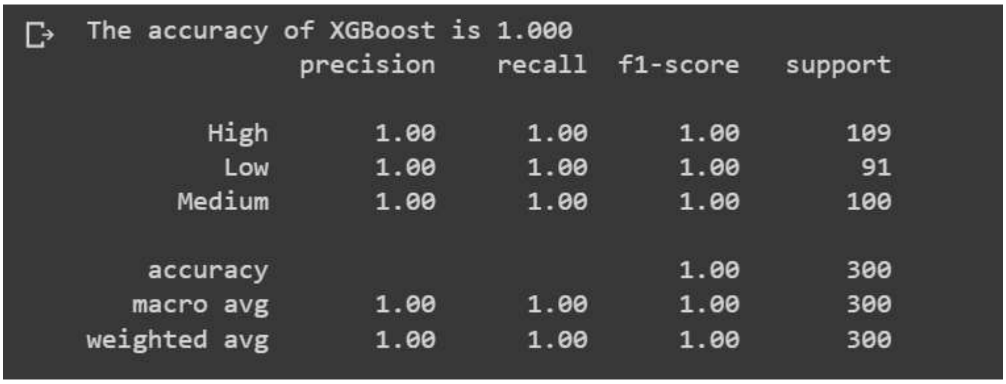
```
The accuracy of KNN is 0.983
              precision    recall  f1-score   support

        High       1.00      1.00      1.00       109
         Low       0.95      1.00      0.97        91
      Medium       1.00      0.95      0.97       100

    accuracy                           0.98       300
   macro avg       0.98      0.98      0.98       300
weighted avg       0.98      0.98      0.98       300
```

Figure 5. Modeling result with KNN (k = 11)

In Figure 5 it can be seen that the accuracy of the KNN model is 98% and the precision, recall, and f1-score values vary but are quite good, this shows that the model built with the KNN algorithm is good and can recognize the given pattern

3.5.2. XGBoost Algorithm

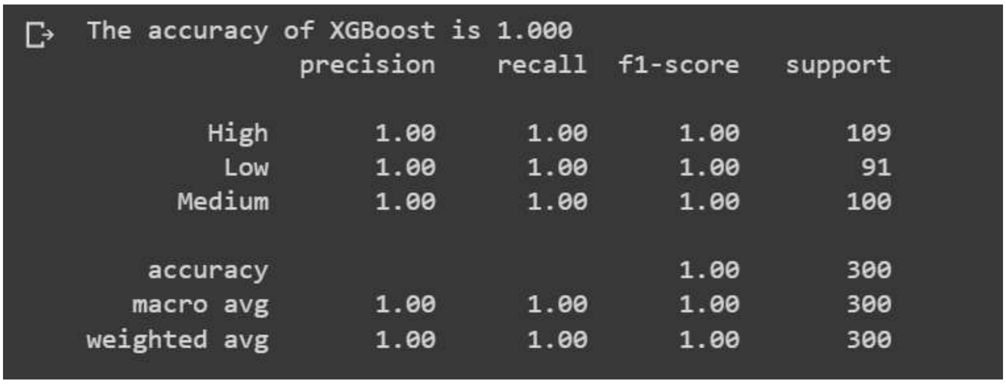
The next modeling stage uses the XGBoost algorithm where this algorithm is also a model for classification as well as KNN. Figure 6 shows the results of the accuracy, precision, recall, and f1-score modeling results from XGBoost with a learning rate = 0.1.



```
The accuracy of XGBoost is 1.000
              precision    recall  f1-score   support

        High       1.00      1.00      1.00       109
         Low       1.00      1.00      1.00        91
      Medium       1.00      1.00      1.00       100

    accuracy                           1.00       300
   macro avg       1.00      1.00      1.00       300
weighted avg       1.00      1.00      1.00       300
```

Figure 6. Modeling result with XGBoost (learning rate = 0.1)

In Figure 6 it can be seen that the accuracy of the XGBoost model is 100% and the precision, recall, and f1-score values are also 100%, this shows that the model built with the XGBoost algorithm is good and can recognize the given pattern. Next, a trial will be carried out with learning rate = 0.5, the results are shown in Figure 7.



```
The accuracy of XGBoost is 1.000
              precision    recall  f1-score   support

        High       1.00      1.00      1.00       109
         Low       1.00      1.00      1.00        91
      Medium       1.00      1.00      1.00       100

    accuracy                           1.00       300
   macro avg       1.00      1.00      1.00       300
weighted avg       1.00      1.00      1.00       300
```

Figure 7. Modeling result with XGBoost (learning rate = 0.5)

In Figure 7 it can be seen that the accuracy of the KNN model is 100% and the precision, recall, and f1-score values are also 100%, this shows that the model built with the XGBoost algorithm is good and can recognize the given pattern.

3.6. Discussion

The testing process was carried out several times using variations in the value of k in the KNN algorithm, and variations in the learning rate in the XGBoost algorithm. The test results can be seen in Figure 8.

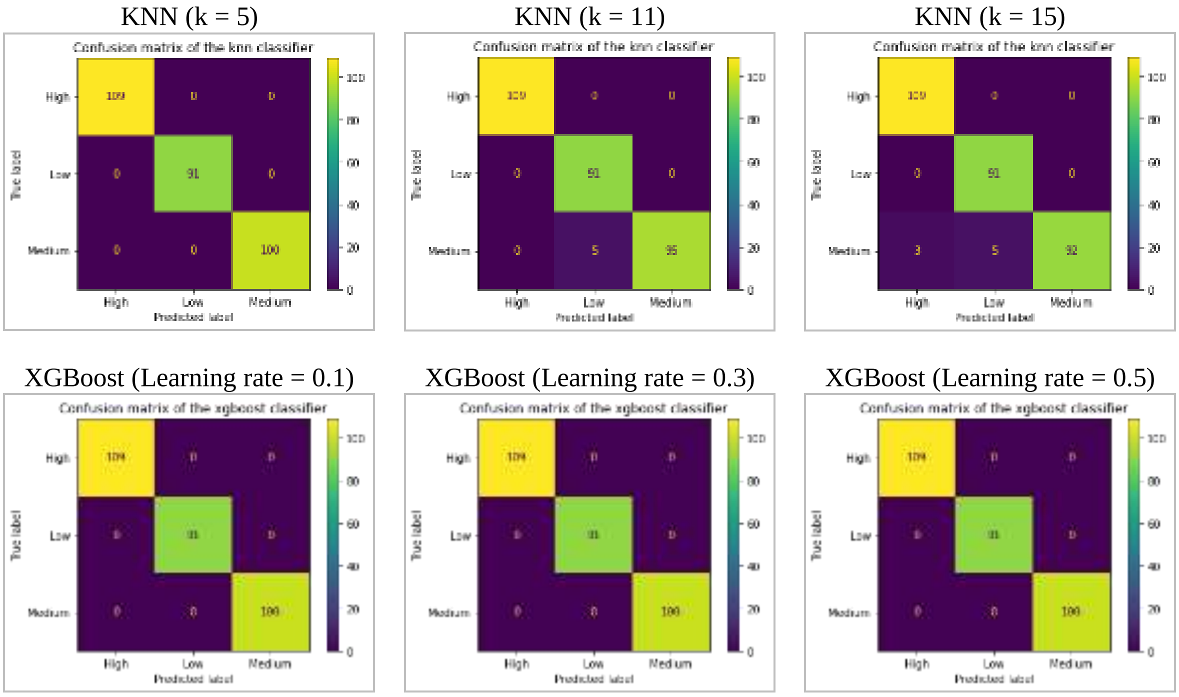

KNN (k = 5)
KNN (k = 11)
KNN (k = 15)
XGBoost (Learning rate = 0.1)
XGBoost (Learning rate = 0.3)
XGBoost (Learning rate = 0.5)

Figure 8. Model test results

In Figure 8 it can be seen that the KNN algorithm has a decreased level of accuracy at a certain k value, in this case the greater the value of k, the level of accuracy will decrease. Meanwhile for the XGBoost algorithm, the level of accuracy of the test tends to be stable even though variations are made on the learning rate value.

4. CONCLUSION

Based on the results of the analysis and testing of the k-Nearest Neighbor algorithm and the XGBoost algorithm for lung cancer prediction, the results show that both algorithms are able to obtain a very good level of accuracy, as well as obtain a balanced precision, recall, and f1-score. However, the KNN algorithm is very dependent on the value of k, so that at a certain value of k, the accuracy of the test will decrease. While the level of accuracy of the XGBoost algorithm tends to be stable even though the learning rate varies. From the results of this research on lung cancer prediction cases, it can be concluded that the XGBoost algorithm tends to be better than the KNN algorithm in recognizing a given pattern, because the modeling process with the XGBoost algorithm is more complex than the KNN algorithm.

REFERENCES

Adege, A. B., Yayeh, Y., Berie, G., Lin, H.-p., Yen, L., & Li, Y. R. (2018). Indoor localization using K-nearest neighbor and artificial neural network back propagation algorithms. *27th Wireless and Optical Communication Conference (WOCC)*. doi:10.1109/WOCC.2018.8372704

American Cancer Society. (2017). Cancer Facts and Figures 2017. *Genes and Development*.

Cherif, I. L., & Kortebi, A. (2019). On Using Extreme Gradient Boosting (XGBoost) Machine Learning Algorithm For Home Network Traffic Classification. *IFIP Wireless Days*, 1-6.

Gu, J., Chen, R., & Wang, S. M. (2022). Prediction models for gastric cancer risk in the general population: a systematic review. *Cancer Prevention Research*, 309-318.

Jaafar, H., Mukahar, N., & Ramli, D. (2016). Methodology of Nearest Neighbor: Design and Comparison of Biometric Image Database. *IEEE Student Conference on Research and Development (SCOReD)*, 1-6.

Lei, W., & Zhaowei, L. (2017). Research on the Humanlike Trajectories Control of Robots Based on the K-Nearest Neighbors. *IEEE Chinese Automation Congress (CAC) : 7746-7751.*

Nardini, C. (2020). Machine learning in oncology: a review. *Ecancermedicalscience*, 1-8.

Osman, A. I., Ahmed, A. N., Chow, M. F., Huang, Y. F., & El-Shafie, A. (2021). Extreme gradient boosting (XGBoost) model to predict the groundwater levels in Selangor Malaysia. *Ain Shams Engineering Journal*, 1545-1556.

Pan, D., Zhao, Z., Zhang, L., & Tang, C. (2017). Recursive Clustering K-Nearest Neighbors Algorithm and the Application in the Classification of Power Quality Disturbance. *IEEE Conference on Energy Internet and Energy System Integration (EI2) : 1-5.*

Sharma, A., & Rani, R. (2021). A systematic review of applications of machine learning in cancer prediction and diagnosis. *Archives of Computational Methods in Engineering*, 4875-4896.

Sinaga, D. C., Tulus, & Sihombing, P. (2020). Performance of distance-based k-nearest neighbor classification method using local mean vector and harmonic distance. *IOP Conference Series: Materials Science and Engineering*, 1-7.

Wayahdi, M. R., Syahputra, D., & Ginting, S. H. (2020). Evaluation of the K-Nearest Neighbor Model with K-Fold Cross Validation on Image Classification. *INFOKUM, 9*(1), 1-6.

Wayahdi, M. R., Tulus, & Lydia, M. S. (2020). Combination of k-means with naïve bayes classifier in the process of image. *IOP Conf. Series: Materials Science and Engineering* (pp. 1-7). IOP Publishing. doi:0.1088/1757-899X/725/1/012126

Wu, H., Yang, S., Huang, S., He, J., & Wang, X. (2018). Type 2 diabetes mellitus prediction model based on data mining. *Inform. Me. Unlocked*, 100-107.

Yuan, M., Zhao, Y., Arkenau, H.-T., Lao, T., Chu, L., & Xu, Q. (2022). Signal pathways and precision therapy of small-cell lung. *Signal Transduction and Targeted Therapy*, 1-18.

Zhang, D., Chen, H.-D., Zulfiqar, H., Yuan, S.-S., Huang, Q.-L., Zhang, Z.-Y., & Deng, K.-J. (2021). iBLP: An XGBoost-Based Predictor for Identifying Bioluminescent Proteins. *Computational and Mathematical Methods in Medicine*, 1-15.