Comparison of classifiers for lung cancer prediction*

Kamil Jędrzkiewicz[1,*,†], Adam Kaszubowski[1,†] and Mateusz Goik[1,†]

[1]Faculty of Applied Mathematics, Silesian University of Technology, Kaszubska 23, 44100 Gliwice, POLAND

Abstract

In this article, we present the program we have developed for lung cancer detection. For making predictions, it uses comprehensive patient information, including gender, age, smoking habits, yellow fingers, anxiety, peer pressure, chronic disease, fatigue, presence of allergies, wheezing, alcohol consumption, coughing, shortness of breath, difficulty swallowing, and chest pain. We start by providing a thorough analysis of the database to identify which features have the most significant impact on the likelihood of developing lung cancer. This includes statistical evaluations and visualizations to better understand the data distribution and correlations between various attributes and lung cancer incidence. Next, we present the results of implementing several different classifiers on the dataset. Through this comparative analysis, we demonstrate that, after preliminary tests, the naive Bayes algorithm emerges as the most effective classifier. We provide the pseudocode for the naive Bayes algorithm, offering a clear and accessible explanation of its implementation. Additionally, we conduct a detailed analysis of its effectiveness, supported by charts and graphs that illustrate the algorithm's accuracy and other relevant performance metrics. Furthermore, we highlight the process of feature selection. By removing irrelevant from the database, we are able to enhance the program's speed and accuracy.

Keywords

Lung cancer, Disease detection, Naive Bayes algorithm, Healthcare

1. Introduction

Lung cancer remains one of the most lethal forms of cancer worldwide[1]. It is difficult to detect in its early stages because its symptoms are very subtle[2]. Fortunately, thanks to the advancements in machine learning algorithms we are now able to improve early detection and diagnosis of this disease to improve patient outcomes. This approach has already worked well with several other kinds of sicknesses such as heart diseases[3], diabetes[4], prostate cancer[5] and breast cancer[6]. In this article, we introduce a cutting-edge program developed for the detection of lung cancer, leveraging the capabilities of machine learning. Utilizing a wide range of patient information—such as gender, age, smoking habits, and other health indicators. Our program employs a naive Bayes algorithm to predict the likelihood of lung cancer with notable accuracy.

This study provides an in-depth analysis of the data features that significantly influence lung cancer risk, offering insights into their relevance and impact. We compare the performance of various classifiers and demonstrate why the naive Bayes algorithm stands out as the most effective after initial testing.[7] Detailed pseudo-code and performance metrics are presented to elucidate the algorithm's efficiency and robustness.

*IVUS2024: Information Society and University Studies 2024, May 17, Kaunas, Lithuania

[1,*] Corresponding author

[†] These author contributed equally.

kj307872@student.polsl.pl (K. Jędrzkiewicz); adamkas324@student.polsl.pl (A. Kaszubowski); mg307866@student.polsl.pl (M. Goik)

0000-0000-0000-0000 (K. Jędrzkiewicz); 0000-0000-0000-0000 (A. Kaszubowski); 0000-0000-0000-0000 (M. Goik)

Furthermore, we explore the process of refining the dataset by eliminating unnecessary information, which enhances both the speed and accuracy of the predictions. This article not only showcases the technical aspects of our program but also emphasizes its potential to revolutionize lung cancer diagnosis, offering a valuable tool for healthcare professionals in the fight against this devastating disease.

2. Methodology

In order to choose the classifier that best suits our task, we have conducted a test of three popular algorithms: k nearest neighbours classifier, naive Bayes classifier and decision tree classifier. Each of the algorithms has been run 500 times, each time with random training and test dataset. Then, we have calculated mean accuracy for all of classifiers and compared their results.

2.1. KNN Classifier

KNN (k-nearest neighbors) is one of the most basic and popular classification algorithm. It measures the distance between the new sample and all points in the training set, identifies the K nearest neighbors, and assigns the most common class label among these neighbors to the new sample.[8] In our project, we used the Euclidean metric to calculate the distance.
We tested for k=2,3,4,5,6,7 and the best was k = 3 and k = 5, with k = 2 being by far the worst. 2

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

where:

- x_i: The i-th coordinate of the point **x**.
- y_i: The i-th coordinate of the point **y**.

2.2. Naive Bayes Classifier

The Gaussian Naive Bayes classifier works by classifying a sample based on the probabilities of each class given the feature values, assuming that features follow a Gaussian (normal) distribution. It calculates the likelihood of the sample's features for each class, combines these with the prior probabilities of the classes, and assigns the class with the highest resulting probability to the sample.
We decided on the Gaussian Naive Bayes because it had the highest efficiency 2
The formula for the conditional probability of a feature x_i given class y is:

$$P(x_i|y) = \frac{1}{\sqrt{2\pi\sigma_{i,y}^2}} \exp\left(-\frac{(x_i - \mu_{i,y})^2}{2\sigma_{i,y}^2}\right)$$

where:

- $P(x_i|y)$: The conditional probability of feature x_i given class y.
- $\sigma_{i,y}$: The standard deviation of feature x_i in class y. It measures the spread of the feature values around the mean.
- $\mu_{i,y}$: The mean (average) of feature x_i in class y. It represents the central value of the feature for the given class.
- x_i: The value of the i-th feature.
- y: The class label.

Algorithm 1: Gaussian Naive Bayes

Data: training data, object to classify
Result: class to which the object belongs

```
groups = split training data into groups according to their class;
best_class = "";
best_score = 0;
for group in groups do
    score = log(number of rows in group/number of rows in all training data);
    for column in group do
        std = standard deviation for column;
        mean = mean for column;
        x = value of column from object to classify;
        col_score = 1/(std·√(2π)) exp(−(x−mean)²/(2·std²));
        score += log(col_score);
    if score > best_score then
        best_score = score;
        best_class = class of group group;
return best_class
```

2.3. Decision Tree Classifier

The Decision Tree classifier works by recursively splitting the dataset into subsets based on feature values, creating a tree structure where each node represents a feature and each branch represents a decision rule. It continues splitting until the subsets are as pure as possible, meaning they contain samples predominantly from one class. The class label assigned to a new sample is determined by traversing the tree according to the sample's feature values until reaching a leaf node, which represents the predicted class. [9]

3. Experiments

3.1. Dataset Description

Our database consists of 16 columns and 309 rows. Individual information includes information about the patient such as gender, age, smoking, yellow fingers, anxiety, peer pressure, chronic disease, fatigue, allergy, wheezing, alcohol consuming, coughing, shortness of breath, swallow- ing difficulty, chest pain and lung cancer, which tells us whether the person has cancer. A value of 1 means that the patient does not have a given symptom and 2 means that he does.
We made a correlation matrix. We were most interested in the last row to find out which symptoms have a positive correlation with lung cancer. From it we can conclude that smoking and shortness of breath have the lowest correlation (but still positive) while allergy and alcohol consuming have the highest correlation.

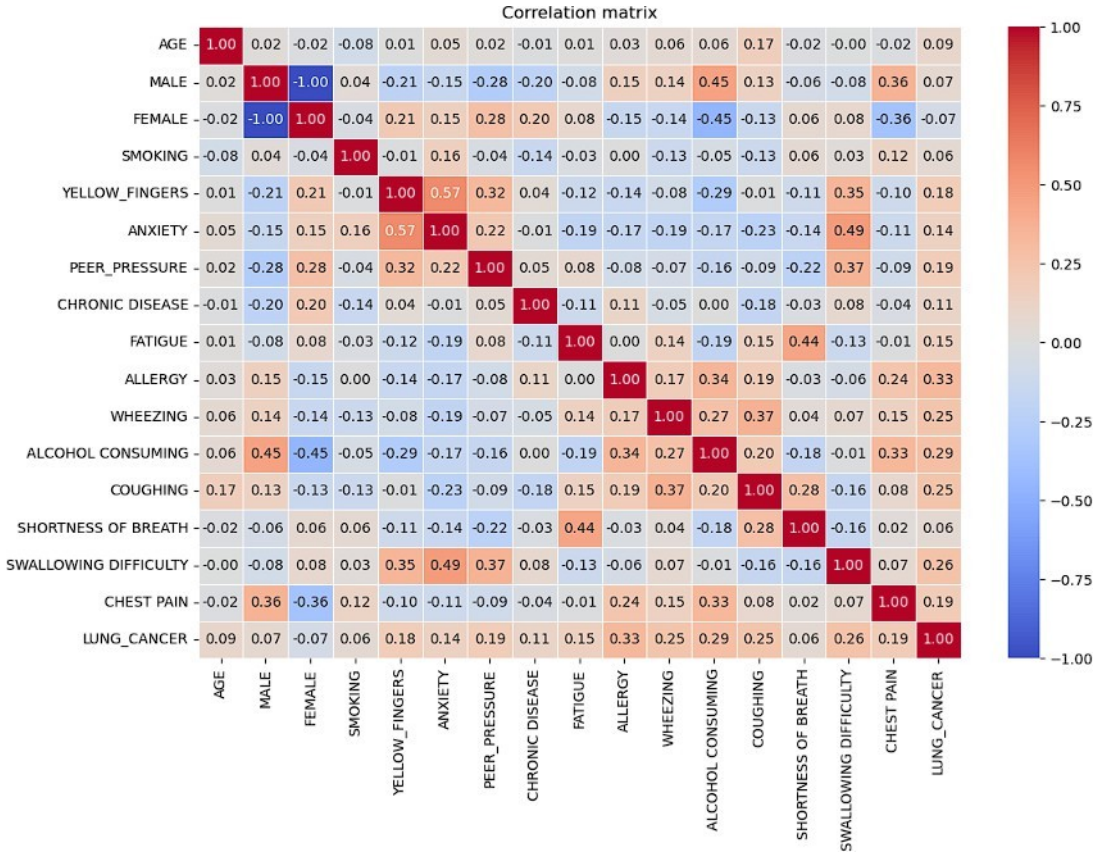

Figure 1: Correlation matrix

3.2. Testing

We compared 4 classifiers to check which one would work best for our data. We used K Nearest Neighbours, Decision Tree, Gaussian Naive Bayes and Multinomial Naive Bayes. As you can see in the figure 2, the Gaussian Naive Bayes has the highest accuracy.

```
Accuracy KNN (k=2) is = 83.42% and standard deviation: 3.39
Accuracy KNN (k=3) is = 87.47% and standard deviation: 3.01
Accuracy KNN (k=4) is = 87.07% and standard deviation: 3.03
Accuracy KNN (k=5) is = 87.62% and standard deviation: 3.06
Accuracy Naive Bayes (Gauss) is = 89.98% and standard deviation: 3.44
Accuracy Naive Bayes (Multinomial) is = 87.31% and standard deviation: 2.93
Accuracy Decision Tree is = 87.7% and standard deviation: 3.33
```

Figure 2: Algorithms accuracy

We then removed one column and checked how its removal would affect the accuracy of the classifier. The differences were negligible, so we decided to remove several columns at once. The best results were obtained after removing columns such as: 'WHEEZING', 'SWALLOWING DIFFICULTY', 'AGE', 'COUGHING', 'SMOKING' where the accuracy of the model averaged 91.28%, and for the best sample of 500 was 100%

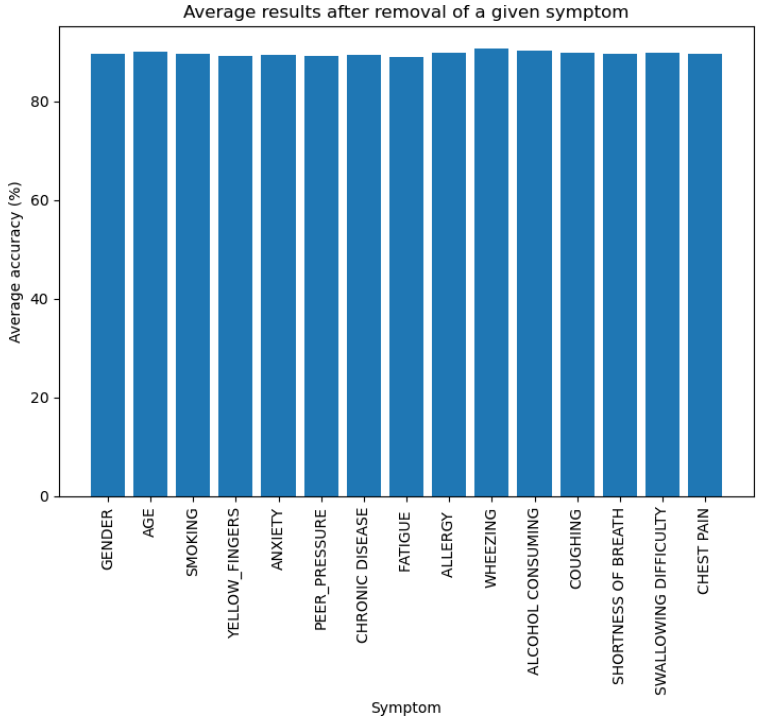

Figure 3: Average results after removal of a given symptom

3.3. Results Analysis

We also created an error matrix for each classifier and calculated: Accuracy, Recall, Precision, F1 and Specificity[10].

The following values were calculated from the formulas:

- Accuracy - determines what part of all classified texts was classified correctly

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

- Recall - determines the share of correctly predicted positive cases (TP) among all positive cases

$$\text{Recall} = \frac{TP}{TP + FN}$$

- Precision - determines how many of the examples predicted positively are actually positive

$$\text{Precision} = \frac{TP}{TP + FP}$$

- F1 - is the harmonic mean between precision and recall. The closer it is to one, the better it proves about the classification algorithm.

$$\text{F1} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

- Specificity - determines how often the model accurately predicted falsehood when something was actually false

$$\text{Specificity} = \frac{TN}{TN + FP}$$

The meaning of symbols:

- TP - the sick person was correctly classified
- TN - a healthy person has been correctly classified
- FP - the sick person was classified as healthy
- FN - a healthy person has been classified as sick

Table 1
Analyze Results

Classifier	Accuracy	Recall	Precision	F1	Specificity
KNN(5)	0.89	0.98	0.90	0.94	0.18
Gaussian Naive Bayes	0.90	0.93	0.94	0.94	0.55
Multinomial Naive Bayes	0.88	1.00	0.88	0.94	0.00
Decision Tree	0.88	0.93	0.94	0.94	0.55

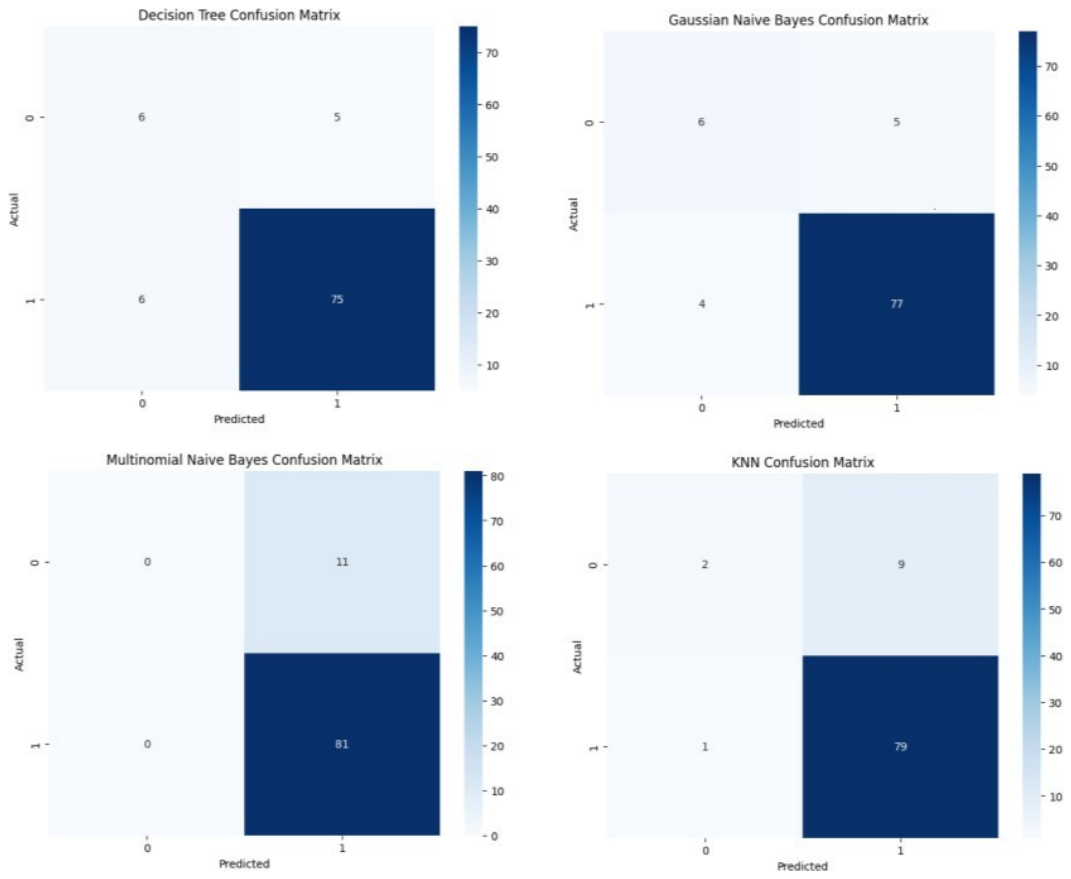

Figure 4: Confusion matrix

4. Conclusion

In conclusion, our study presents a novel approach to lung cancer detection through the integration of machine learning algorithms and comprehensive patient data analysis.

Our research highlights the importance of feature selection in optimizing algorithm performance, leading to improved prediction accuracy and efficiency. Through comparative analysis and detailed evaluation, we have demonstrated the superiority of the naive Bayes algorithm in this context.

By facilitating early detection and intervention, our approach has the potential to significantly improve patient outcomes and contribute to the ongoing efforts to combat this deadly disease.

References

[1] J. Malhotra, M. Malvezzi, E. Negri, C. La Vecchia, P. Boffetta, Risk factors for lung cancer worldwide, European Respiratory Journal 48 (2016) 889–902.

[2] R. L. Krech, J. Davis, D. Walsh, E. B. Curtis, Symptoms of lung cancer, Palliative Medicine 6 (1992) 309–315. URL: https://doi.org/10.1177/026921639200600406. doi:10.1177/026921639200600406. arXiv:https://doi.org/10.1177/026921639200600406.

[3] H. Arghandabi, P. Shams, A comparative study of machine learning algorithms for the prediction of heart disease, International Journal for Research in Applied Science and Engineering Technology 8 (2020) 677–683.

[4] A. Mujumdar, V. Vaidehi, Diabetes prediction using machine learning algorithms, Procedia Computer Science 165 (2019) 292–299. URL: https://www.sciencedirect.com/science/article/pii/S1877050920300557. doi:https://doi.org/10.1016/j.procs.2020.01.047, 2nd International Conference on Recent Trends in Advanced Computing ICRTAC -DISRUP - TIV INNOVATION , 2019 November 11-12, 2019.

[5] M. M. I. Molla, J. Jui, H. Rana, N. Podder, Machine Learning Algorithms for the Prediction of Prostate Cancer, 2023, pp. 471–482. doi:10.1007/978-981-19-7528-8_37.

[6] M. Amrane, S. Oukid, I. Gagaoua, T. Ensarİ, Breast cancer classification using machine learning, in: 2018 Electric Electronics, Computer Science, Biomedical Engineerings' Meeting (EBBT), 2018, pp. 1–4. doi:10.1109/EBBT.2018.8391453.

[7] E. M. E. F. Christian Dwi Suhendra, Effan Najwaini, A machine learning perspective on daisy and dandelion classification: Gaussian naive bayes with sobel, Indonesian Journal of Data and Science 4 (2023) 151–159.

[8] X. Mu, Implementation of music genre classifier using knn algorithm, Highlights in Science Engineering and Technology 34 (2023) 149–154.

[9] V. V. Karnika Dwivedi, Hari Om Sharan, Analysis of decision tree for diabetes prediction, International Journal of Engineering and Technical Research (IJETR) 9 (2019) 3–6.

[10] B. Juba, H. S. Le, Precision-recall versus accuracy and the role of large data sets, in: Proceedings of the AAAI conference on artificial intelligence, volume 33, 2019, pp. 4039–4048.