Mortality Prediction: A Naïve Bayes Classification Approach

Abdallah Magdy
Systems and Biomedical Engineering
Cairo University

Muhammad Alaa
Systems and Biomedical Engineering
Cairo University

Osama Muhammad
Systems and Biomedical Engineering
Cairo University

Ziad Ahmed
Systems and Biomedical Engineering
Cairo University

Aya Eyad
Systems and Biomedical Engineering
Cairo University

***Abstract*—This paper discusses the implementation of a Gaussian Naïve Bayes classifier to implement a mortality prediction model.**

Keyword—Naïve Bayes, Machine learning, statistical testing, mortality prediction

I. INTRODUCTION

Accurate prediction of mortality is of great importance in public health research, enabling proactive healthcare planning, resource allocation, and policy development. Mortality prediction models that incorporate relevant predictors can assist healthcare providers and policymakers in identifying high-risk populations, implementing targeted interventions, and improving overall healthcare outcomes. While several approaches have been explored in mortality prediction, the integration of health indicators and socioeconomic factors has shown promise in capturing the complex relationship between individual health characteristics and mortality risk.

This research paper aims to develop a mortality prediction model using a Gaussian Naive Bayes classification approach that incorporates various health indicators. Naive Bayes is a well-established and computationally efficient classification algorithm that assumes independence between predictors, Gaussian Naive Bayes also assumes the features distributions are Gaussian.

II. THEORETICAL BACKGROUND

A. Gaussian Naïve Bayes' Classifier

A Naive Bayes classifier is a simple probabilistic classifier based on applying Bayes' theorem with strong (naive) independence assumptions.

The Gaussian classifier assumes the data is described by a Gaussian distribution.

The Naïve Bayes classifiers are based on Bayes' Theorem which states that

$$P(Y|X) = \frac{P(X|Y)\,P(Y)}{P(X)}$$

where $P(Y)$ is the prior probability of event Y

$P(Y|X)$ is the probability of event Y given the event X

$P(X|Y)$ is the probability of event X given the event Y

$P(X)$ is the prior probability of event Y

X represents a vector of features. That's why the independence of feature assumption is of great importance.

$$P(Y|X) = \frac{\prod_{i=1}^{i=n} P(x_i|Y)\,P(Y)}{P(X)}$$

where X is the vector of features $(x_1, x_2, \ldots, x_n)$

x_i is an individual feature

n is the number of features

Y is the event someone would live for the next 10 years

To select the class (label) with the highest probability:

$$Y = argmax_Y P(Y|X) = argmax_y \frac{P(x_i|Y)P(Y)}{P(X)}$$

Since P(X) does not depend on the posterior probability we can neglect it when look for the correct class.

$$Y = argmax_y P(Y|X) = argmax_y (P(x_i|Y)P(Y))$$

Because the values or probabilities are between 0 and 1, their multiplication would result in a very small number. So, we will apply the log function which will also change the multiplication into summation.

$$Y = argmax_Y (\sum_{i=1}^{i=n} log(P(x_i|Y))) + log(P(Y))$$

Finally we will calculate P(Y) as the frequency of each class in our dataset and $P(x_i|Y)$ is the class conditional probability which is modelled with the Gaussian distribution.

$$P(x_i|Y) = \frac{e^{\frac{-(x_i-\mu_Y)^2}{2\,{\sigma_Y}^2}}}{\sqrt{2\pi{\sigma_Y}^2}}$$

III. METHODS

This study started by choosing the suitable dataset, followed by data cleaning of null values, duplicates, zeroes and outliers using the IQR method. Followed by the data preprocessing to fit the Gaussian Naïve Bayes assumptions of normality, graphically using QQ plots and using statistical tests in particular the Anderson-Darling's test, and independence of features. Followed by the GNB model development and comparison with the built-in GNB model of the Scikit Learn library. The last step was the development of an Adaboost model, a Random Forest model and an ensemble of models to compare with GNB and yield a higher accuracy. These steps are described in detail in the following sections.

The used software packages to conduct this study are Pandas, NumPy, Matplotlib, Seaborn, Scipy, Sklearn, Missingno, Mpld3, Shap and Statsmodel.

IV. DATASET DESCRIPTION

The used dataset [1] is the NDI (National Death Index) Mortality Data, The NCHS (National Center for Health Statistics) has linked data from various surveys with death certificate records from the National Death Index (NDI).

Some of the surveys are:

- National Health Interview Survey (NHIS): 1986 – 2018
- National Health and Nutrition Examination Surveys (NHANES): 1999-2018
- Third National Health and Nutrition Examination Survey (NHANES III)
- NHANES I Epidemiologic Follow-up Study (NHEFS)
- National Nursing Home Surveys (NNHS): 1985, 1995, 1997, 2004
- The Second Longitudinal Study of Aging (LSOA II)

The dataset consists of 8579 records, 18 features and one label column.

A. The Data Features

The features range from health indicators to socioeconomic factors.

The **health indicators** are:

1. **Systolic, Diastolic and Pulse Blood Pressures** [2]

Normal levels of both systolic and diastolic blood pressure are particularly important for the efficient function of vital organs such as the heart, brain and kidney and for overall health and well-being. Pulse pressure is also an indicator for the risk of a heart event like a heart attack or stroke.

Systolic BP

It is the upper number in the blood pressure reading, indicates how much pressure blood is exerting against the artery walls when the heart contracts, measured in mm Hg.

Diastolic BP

It is the lower number in the blood pressure reading, indicates how much pressure blood exerts against the artery walls while the heart muscle rests between contractions, measured in mm Hg.

Pulse pressure

The difference between the systolic and diastolic blood pressures. It represents the force that the heart generates each time it contracts, measured in mm Hg.

2. Red blood cells

Represents the total number of RBCs in the blood measured in 10^{11}/liter. It is necessary for the adequate oxygenation of body tissues.

3. White blood cells

Represents the total number of WBCs in the blood measured in 10^9/liter. It can act as a sign of infection, inflammation or an autoimmune disease.

4. Sedimentation rate

It measures how quickly erythrocytes, or red blood cells, separate from a blood sample that has been treated so the blood will not clot. This blood test can reveal inflammatory activity in the body. It is measured in mm/hr.

5. Serum Albumin

Albumin is a protein made by the liver. A serum albumin test measures the amount of this protein in the clear liquid portion of the blood. It may suggest a problem with the liver or kidneys. It may also indicate that a person has a nutrient deficiency. It is measured in g/dl.

6. Serum Cholesterol

Represents the amount of total cholesterol in the blood. A person's serum cholesterol level can indicate their risk of developing conditions such as heart disease. It is measured in mg/dL.

7. Serum Iron

Represents how much iron is in the blood in mcg/dL. Abnormally low iron levels can indicate anaemia or gastrointestinal blood loss and abnormally high iron levels can indicate liver conditions or hemolytic anemia.

8. Serum Magnesium

It is measured in mg/dL. Measures the magnesium level in blood. Hypermagnesemia can be a sign of kidney failure. Hypomagnesemia can be attributed to chronic disease, alcohol use disorder, gastrointestinal losses, renal losses, and other conditions.

9. Serum Protein

Represents the total amount of protein in the blood, mainly the two major groups of proteins: albumin and globulin blood measured in g/dl. Albumin carries medicines and other substances through the blood and is important for tissue growth and healing. As for globulin, certain globulins bind with hemoglobin, other globulins transport metals, such as iron, in the blood and help fight infection.

10. TIBC

A total iron-binding capacity (TIBC) test measures the blood's ability to attach itself to iron and transport it around the body. It used for the diagnosis of iron deficiency anemias and other disorders of iron metabolism. Measured in mcg/dL.

11. TS

Transferrin Saturation (TS) indicates the percentage of iron binding sites on transferrin that are carrying iron. High transferrin signifies low iron, which means there is less iron bound to transferrin, allowing for a high circulation of non-bound iron transferrin in the body, revealing a possible iron deficiency anemia. It is measured in mg/dL.

12. BMI

It is a value derived from the mass (weight) and height of a person. Expressed in units of kg/m^2. It is a convenient rule of thumb used to broadly categorize a person as underweight, normal weight, overweight, or obese based on tissue mass (muscle, fat, and bone) and height. BMIs under 20 and over 25 have been associated with higher all-cause mortality, with the risk increasing with distance from the 20–25 range. Among people who have never smoked, overweight/obesity is associated with 51% increase in mortality compared with people who have always been a normal weight. [3]

The **socioeconomic indicators** are:

1. Age

A numerical representation of an individual's age.

2. Sex

Biological categorization of individuals as male or female. (1: Male, 2: Female)

3. Race

Categorization based on ethnic or racial backgrounds, represented by specific values for different groups. (1: Non-Hispanic White, 2: Non-Hispanic Black or African American, 3: Mexican American or Hispanic)

4. Poverty Index

A measure of poverty levels within a population or area, considering socioeconomic factors. It provides a quantitative measure of poverty, allowing researchers, policymakers, and organizations to understand and track poverty levels over time and across different regions.

The **categorical** features are sex, race and death (the label). All other features are **quantitative**.

B. *Probability distributions*

We check for the type of probability distribution by plotting a histogram of each feature in our data. The curves shown in the figure are kernel density estimates which are approximations to the probability density function of the distribution. Upon looking at the produced distributions, we deduce that:

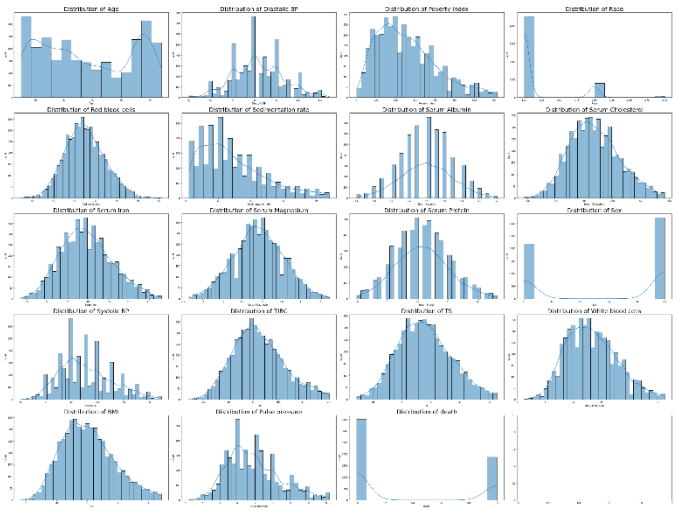

Figure 1: Feature Distributions before transformations

- Age, Diastolic BP, Systolic BP, Pulse Pressure, Sedimentation rate distributions: unknown
- Poverty Index, White Blood Cells, BMI: These variables appear to have a right-skewed normal distribution.
- Sex, Race: categorical variables

The rest appears to follow a normal distribution, but further verification will be conducted later.

C. *Measures of central tendency and dispersion*

The measures of central tendency as the mean, mode and the different quartiles of the data were calculated before and after outlier-removal, as well as the measures of dispersion as the variance, standard deviation, range and IQR (Interquartile Range).

V. DATA PREPARATION

This process includes multiple steps, starting by cleaning the data to remove any existing null values, zeroes or duplicates. Then testing and manipulating the data to fit the two assumptions of Gaussian Naïve Bayes models which are normality and independence, this second step includes applying transformations, standardization and checking for independence.

A. *Data Cleaning*

No null values, zeroes or duplicates were found. As for outliers, a lot of them were found, this can be shown by observing the box plots.

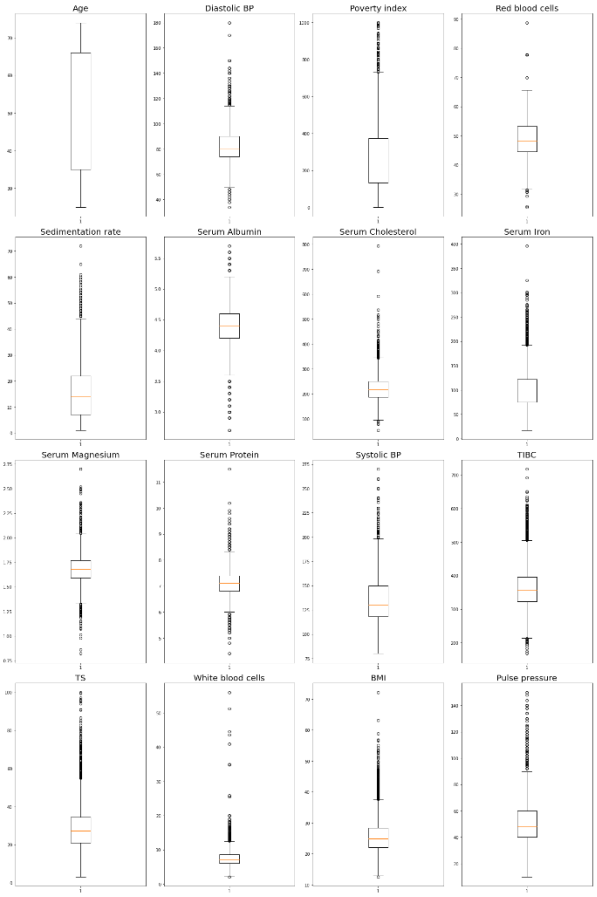

Figure 2: Box plots to show outliers

Two methods were applied, The IQR (Interquartile Range) method and the Modified z-score method

1) *The IQR Method*

$$IQR = Q_3 - Q_1$$
$$Upper\ Threshold = Q_3 + 1.5 \times IQR$$
$$Lower\ Threshold = Q_1 - 1.5 \times IQR$$

where Q_1 is the first quantile,
Q_3 is the third quantile

By removing any data point higher than the upper threshold and lower than the lower threshold. This is the method we choose to continue our analysis.

2) *The Modified Z-score Method*

Knowing that our data is roughly normally distributed, we can use the Z-score method, in which we would consider points to be outliers based on how much they deviate from the mean value; However, the mean is not a robust statistic; it is heavily influenced by outliers, meaning that the outliers we are trying to detect would affect the method itself. So, instead of using the mean and standard deviation, we use the median and the deviation from the median. The median is a robust statistic, meaning it will not be greatly affected by outliers. This is called the Robust Z-score method, and instead of using standard deviation, it uses the MAD (Median Absolute Deviation).

We will need to calculate the median of the sample and the MAD, which is calculated using the following equation.

$$MAD = median\{|x_i - \tilde{x}|\}$$

where $\tilde{x}$ is the median of the sample

We then calculate the modified z-score for each point using the following equation

$$M_i = \frac{0.6745(x_i - \tilde{x})}{MAD}$$

0.6745 is the 0.75^{th} quartile of the standard normal distribution to which the MAD converges to.

As a rule of thumb, a score of 3.5 is used as the cut-off value and each data point with a score larger than 3.5 is considered as an outlier.

We decided to proceed with the IQR method in our analysis.

B. *GNB Assumptions conformation manipulation and check*

Feature transformations refer to the process of modifying or manipulating the features (variables) in a dataset to improve their representation, extract meaningful information, or meet certain assumptions required by a particular analysis or model.

Since GNB necessitates that the input data is normal, we tried applying many transformations on our data to reduce skewness and approach normality. The most frequently used transformations are the square root, logarithm and reciprocal transformations. In the following figure is our data distributions after applying the log and square root transformation.

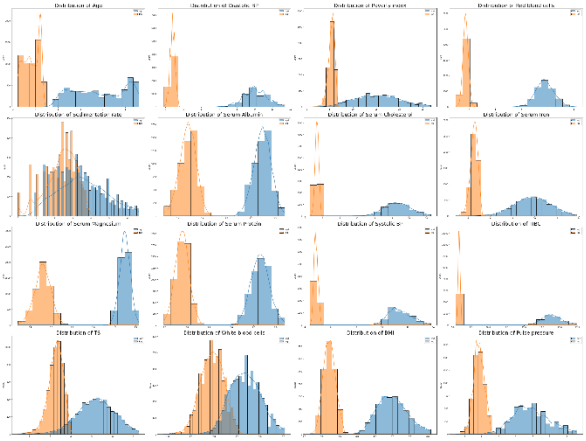

Figure 3: Feature distributions after square root and log transformations

Results were not satisfactory after applying both transformations so we tried another transformation which is the Box cox transformation.

The Box cox is a transformation of on-normal dependent variables into a normal shape, depending on a parameter lambda (λ) which varies from -5 to 5. All values of λ are considered and the optimal value for the data is selected; The optimal value is the one which results in the best approximation of a normal distribution curve. The transformation of Y has the form:

$$y(\lambda)\begin{cases}\frac{y^{\lambda}-1}{\lambda} & if\ \lambda \neq 0\\ \log(y) & if\ \lambda = 0\end{cases}$$

where y is the original data

y(λ) is the transformed data

We can notice from the formula that the log transform is a special case of the Box cox with lambda equal to zero.

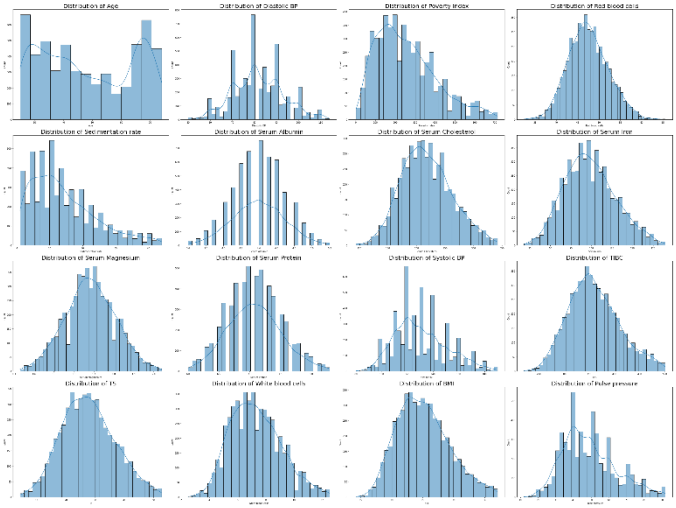

Figure 4: Feature distributions after Box cox transformation

The Box cox Transformation resulted in a good visual approximation for a normal distribution so we proceed with its transformed features.

Next, we have to check for the normality of the features. Several graphical methods and statistical tests can be used when using a model that assumes normality to find whether the sample was taken from a population that follows a normal distribution or not.

An example of the graphical methods used is the QQ (Quantile-Quantile) plot. A QQ plot is a scatterplot created by plotting two sets of quantiles against one another. If both sets of quantiles came from the same distribution, we should see the points forming a line that's roughly straight. It can be used to plot the quartiles of our data points against the quartiles of a real normal distribution. The following figure is the QQ plot for all our features

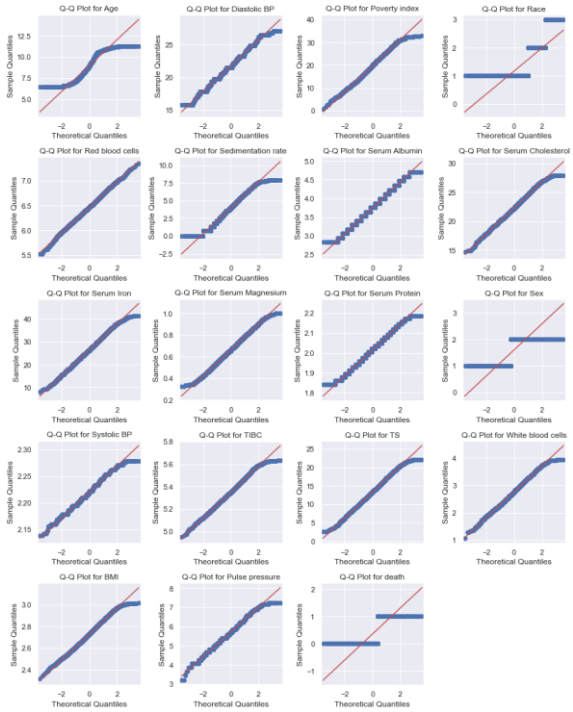

Figure 5: QQ plots

From the figure, we can deduce that the red blood cell column seems to follow the normal distribution pretty closely. Since it is a visual check, not an air-tight proof, we need to support the decision with statistical tests.

Of the existing tests for normality are

- The Shapiro-Wilk test
- The Kolmogorov-Smirnov test
- The Anderson-Darling's test

The Shapiro–Wilk test is a more appropriate method for small sample sizes (<50 samples), so it will not be used in our case (5384 samples after removing outliers).

Both the Kolmogorov-Smirnov and the Anderson-Darling's test can be used for larger samples, we chose to apply the Anderson-Darling's test which is a modification of the Kolmogorov-Smirnov test and gives more weight to the tails than the K-S test.

The Anderson-Darling tests the null hypothesis that a sample is drawn from a population that follows a particular distribution. For the Anderson-Darling test, the critical values depend on which distribution is being tested against. This function works for normal, exponential or Gumbel distributions.

H_0 (The null hypothesis): The sample comes from a normally distributed population.
H_1(The alternative hypothesis): The sample comes from a non-normally distributed population.

Applying the Anderson-Darling test to our data at a significance level of 0.05, the null hypothesis was rejected for all features except for the Red blood cells which conforms with the QQ plots presented earlier.

Next, we will proceed with the check for feature independence. Linear independence can be checked using Pearson's correlation coefficient calculated using the following formula:

$$r = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2 \sum(y_i - \bar{y})^2}}$$

We can visualize the correlation coefficients between different features using heatmaps

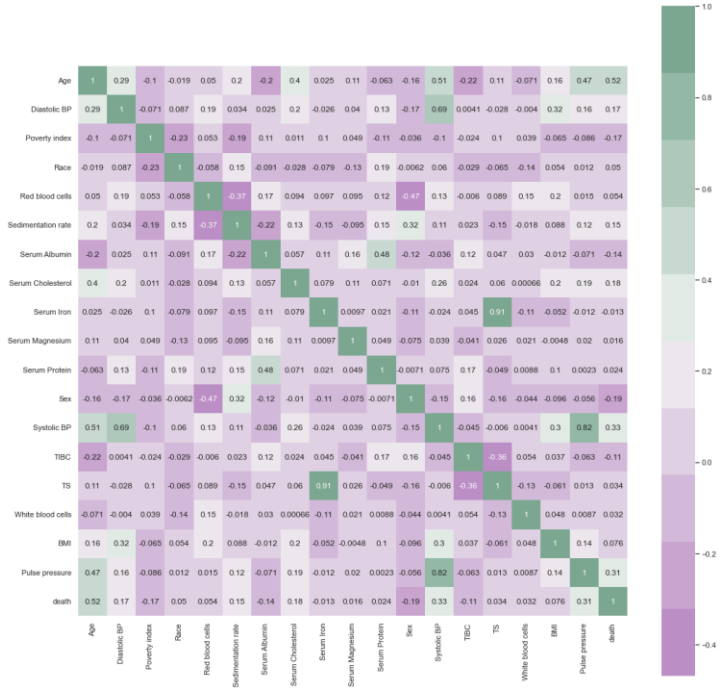

Figure 6: Heatmap to visualize correlation coefficients

We can see that there is a relatively high correlation between the TS and Serum Iron columns and the Systolic BP and Pulse Pressure columns, which indicates that one of these columns should be dropped. However, we tried both for the final model and there was not much of a difference in performance.

C. *Standardization*

Feature standardization is a preprocessing technique used to transform numerical features in a dataset to a common scale. It aims to ensure that all features contribute equally to the analysis and modeling processes by eliminating or reducing the potential bias introduced by differences in the scales or units of the features. Standardization (Z-score normalization) was carried on the transformed data using the following equation.

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

where μ is the mean of the sample feature and σ is the standard deviation of the sample feature.

D. Data Splitting

The transformed and standardized data was split into 80% training data used in training the model and 20% testing data used to calculate the model accuracy and ability to generalize on unseen data.

VI. RESULTS AND ANALYSIS

A. Gaussian Naïve Bayes

As stated earlier, the main formula behind Gaussian Naïve Bayes classification is:

$$Y = argmax_Y(\sum_{i=1}^{i=n} log(P(x_i|Y))) + log(P(Y))$$

Each feature contributes to the decision-making as well as the prior probability. To take a look into the decision-making of our model we will calculate the prior probabilities and plot the conditional probability of each of our features on each target class.

$P(Y = 0)$ is the frequency of subjects in our sample who lived for the following ten years.

$P(Y = 1)$ is the frequency of subjects in our sample who died in the following ten years.

Then, the model compares the conditional probabilities $P(x_i|Y)$ for Y = 0 and Y = 1 calculated using the following formula (since the distributions are close to normal):

$$P(x_i|Y) = \frac{e^{\frac{-(x_i-\mu_Y)^2}{2\,{\sigma_Y}^2}}}{\sqrt{2\pi{\sigma_Y}^2}}$$

We can visualize this process by drawing the conditional probabilities for each feature on each target class.

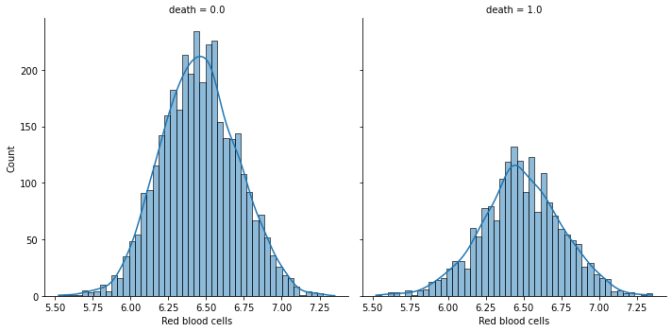

Figure 7: The conditional probability for the RBC feature for each target class

Note that that the values on the x-axis are those of the transformed data not the original data.

Considering the age feature alone, if we were to classify a subject having a red blood cell count of 6.5 after transformation, the conditional probability of P(RBC = 6.5 | death = 0) is higher than the conditional probability of P(RBC = 6.5 | death = 1), so the contribution of the age feature to the decision-making of the model would lean towards classifying the subject as alive.

The class having a higher sum of conditional probability after calculating the conditional probabilities for the whole set of features and adding the prior probabilities will be the chosen label by the model.

Model Results

A) Accuracy

The GNB implemented from scratch produced an accuracy of 77.808% and the GNB of Sklearn produced the same accuracy of 77.808%.

B) Confusion Matrix

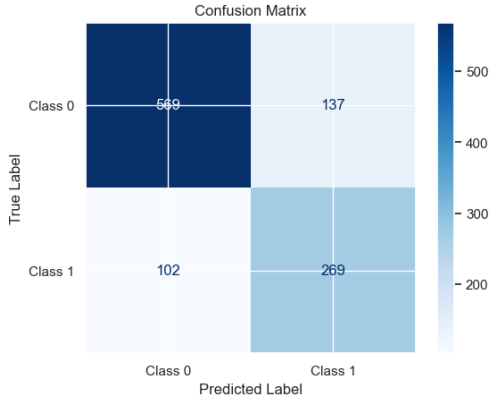

Figure 8: GNB confusion matrix

A confusion matrix is a table that is used to define the performance of a classification algorithm. It shows four combinations of the true and predicted labels. The following figure is the confusion matrix of the built-in GNB model showing the number of true positives, true negatives, false positives and false negatives.

C) ROC Curve

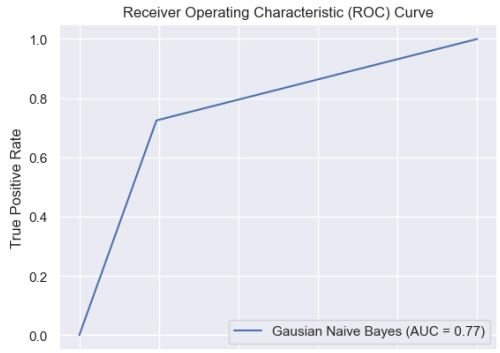

Figure 9: GNB ROC curve

B. *Adaboost Classifier*

AdaBoost (Adaptive Boosting) is a machine learning algorithm used for classification tasks. It combines multiple weak classifiers, such as decision stumps, into a single strong classifier. Initially, all training examples are assigned equal weights. Weak classifiers are trained sequentially, with weights adjusted to focus on misclassified examples. The final classifier is formed by combining the weak classifiers based on their performance, with more accurate classifiers having higher weights. To classify new instances, the weak classifiers' predictions are combined using their assigned weights. Adaboost leverages the concept of boosting to iteratively improve its classification accuracy by emphasizing challenging examples. This is how Adaboost works briefly:

1. Initialization: Each training example denoted as (x_i, y_i), where x_i is the imput features and y_i is the corresponding class label, ia assigned an initial weight $\omega_i = \frac{1}{n}$

   where is the total number of training examples.

2. Training Weak Classifiers: AdaBoost sequentially trains a series of weak classifiers. Each weak classifier, denoted as $h_t(x)$, is trained on the weighted training data, where the weights represent the importance of each example. The weak classifier aims to minimize the weighted error rate, Err_t, defined as the sum of weights of misclassified examples:

   $$Err_t = \Sigma_i(w_i * I(y_i \neq h_t(x_i)))$$

3. Weight Update: Once the weak classifier is trained, its weight, α_t, is calculated based on its performance:

   $$\alpha_t = 0.5 * ln(\frac{1 - Err_t}{Err_t}).$$

   The weight α_t measures the contribution of the weak classifier in the final classification. Higher values of α_t are assigned to more accurate classifiers, while lower values are assigned to weaker ones. The weights of the training examples are updated as follows:

   $$w_i \leftarrow w_i * e^{(-\alpha_t * y_i * h_t(x_i))}$$

   This update increases the weights of the misclassified examples, making them more important for subsequent classifiers to focus on.

4. Combining Classifiers: The final classification is determined by combining the weak classifiers' predictions using their weights. Given a new instance x, the AdaBoost classifier output, H(x), is calculated as

   $$H(x) = sign(\Sigma_t(\alpha_t * h_t(x)))$$

   Here $sign()$ returns the sign of the sum, indicating the predicted class label $(+1\ or\ -1)$.

Model Results

The Adaboost model produced an accuracy of 80.241% using a 30 estimators and a learning rate of 0.4.

C. *Random Forest Classifier Model*

The Random Forest classifier is a popular machine learning algorithm used for classification tasks. It belongs to the ensemble learning family and combines the predictions of multiple decision trees to make accurate predictions. Here's a brief explanation of how Random Forest works:

1. Random Subsampling: The algorithm starts by creating an ensemble of decision trees. Each tree is trained on a random subset of the training data, sampled with replacement (known as bootstrap aggregating or "bagging"). This creates diverse subsets of data for each tree.

2. Feature Randomness: During the construction of each decision tree, a random subset of features is considered for splitting at each node. This introduces further randomness and helps to reduce correlation among the trees.

3. Decision Tree Construction: Each decision tree is constructed by recursively partitioning the data based on the selected features. The splitting is done based on criteria such as Gini impurity or information gain, aiming to create nodes that best separate the classes.

4. Voting for Classification: Once all the trees are built, to classify a new instance, each tree independently predicts the class label. The final prediction is made by majority voting, where the class that receives the most votes across all trees is chosen as the final predicted class.

Model Results

The Random Forest classifier produced an accuracy of 81.244%.

D. *Using an ensemble of models*

Using several different classifiers (MLP Classifier, Logistic Regression, SVC, RF Classifier), and use a voting classifier to detect the best accuracy among all these different classifiers.

Model Results

The ensemble of models produced an accuracy of 81.337%.

Accuracies of the four models

POC	**Model Name**			
	Gaussian Naïve Bayes	***Adaboost***	***Random Forest***	***Ensemble***
Accuracy	77.808%	79.823%	81.244%	81.337%

VII. CONCLUSION

In conclusion, this research paper aimed to explore the Gaussian Naive Bayes algorithm and its performance on the problem of mortality detection and compare it with the

performance of different machine learning models, including Adaboost, Random Forest, and an ensemble of models consisting of MLP Classifier, Logistic Regression, SVC, and RF Classifier. After conducting extensive experiments and analyzing the results, it is evident that the ensemble of models outperformed the individual models in terms of accuracy. The ensemble approach combines the strengths of multiple models, leveraging their diverse perspectives and learning abilities to make more accurate predictions. The ensemble model's superior performance can be attributed to its ability to capture a broader range of patterns and relationships in the data. By aggregating the predictions from multiple models, the ensemble approach mitigates the limitations of individual models and increases the overall accuracy.

MEMBER PARTICIPATION

Data Cleaning: Aya Eyad, Ziad Ahmed
Data Analysis: Muhammad Alaa, Abdallah Magdy
Report: Aya Eyad, Osama Muhammad
Presentation: Abdallah Magdy, Ziad Ahmed
Testing Muhammad Alaa, Ziad Ahmed
Naive Bayes explanation and model implementation: Osama Muhammad, Aya Eyad
Additional models: Muhammad Alaa
Jupyter book: Abdallah Magdy

References

[1] *NCHS data linkage - mortality data* (2023) *Centers for Disease Control and Prevention*. Available at: https://www.cdc.gov/nchs/data-linkage/mortality.htm (Accessed: 24 June 2023).

[2] Association, A.H. (2023a) *Understanding blood pressure readings*, *www.heart.org*. Available at: https://www.heart.org/en/health-topics/high-blood-pressure/understanding-blood-pressure-readings (Accessed: 24 June 2023).

[3] Global BMI Mortality Collaboration None;Di Angelantonio E;Bhupathiraju ShN;Wormser D;Gao P;Kaptoge S;Berrington de Gonzalez A;Cairns BJ;Huxley R;Jackson ChL;Joshy G;Lewington S;Manson JE;Murphy N;Patel AV;Samet JM;Woodward M;Zheng W;Zhou M;Bansal N;Barric (no date) *Body-mass index and all-cause mortality: Individual-participant-data meta-analysis of 239 prospective studies in Four Continents*, *Lancet (London, England)*. Available at: https://pubmed.ncbi.nlm.nih.gov/27423262/ (Accessed: 24 June 2023).

[4] *Smoking and reverse causation create an obesity ... - wiley online library*. Available at: https://onlinelibrary.wiley.com/doi/10.1002/oby.21239 (Accessed: 24 June 2023).