DOI: http://dx.doi.org/10.26483/ijarcs.v15i4.7106

Volume 15, No. 4, July-August 2024

ISSN No. 0976-5697

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Optimizing Fetal health Classification with PCA and SMOTE Techniques

Yashaswini K S
Department of Computer Science & Engineering
Sri Jayachamarajendra College of Engineering (SJCE)
Mysuru, India

Chandana Raju M J
Department of Computer Science & Engineering
Sri Jayachamarajendra College of Engineering (SJCE)
Mysuru, India

Vaishnavi K
Department of Computer Science & Engineering
Sri Jayachamarajendra College of Engineering (SJCE)
Mysuru, India

Meghana P
Department of Computer Science & Engineering
Sri Jayachamarajendra College of Engineering (SJCE)
Mysuru, India

Abstract: Cardiotocography (CTG) is used in pregnancy to monitor fetal heart rate and contractility, especially in the third trimester, to ensure fetal well-being and to detect early signs of distress CTG a inconsistency may indicate the need for further research and possible interventions. The objective is to increase the accuracy and reliability of cervical health classification by integrating machine learning algorithms with traditional CTG data. This approach seeks to improve the early detection of fetal distress to identify timely medical interventions. The system combines CTG data collection with machine learning algorithms to identify fetal health risks. It uses transducers to monitor fetal heart rate and contractions. Machine learning models are used to analyze CTG data, such as random forest, logistic regression, decision tree and KNN, the results showed that the random forest model outperformed the others, achieving an accuracy of 97.58 %.

Keywords: Cardiotocography, fetal heart rate, uterine contractions, machine learning, fetal distress, random forest, prenatal monitoring and medical interventions.

I. INTRODUCTION

A key component of artificial intelligence, machine learning uses "training data" to anticipate outcomes in a variety of applications such as computer vision and email filtering, allowing computers to learn from experience and perform better without explicit programming. Machine learning shines in situations when traditional algorithms are impracticable, enabling autonomous learning from data to complete tasks. Predictive analytics is a branch of advanced analytics that helps predict future occurrences by turning data into insightful information for decision-making through the use of data mining, machine learning, and artificial intelligence. The enormous need for medical information in the healthcare industry outpaces human capacity, necessitating in-depth investigation to incorporate patient-specific characteristics. The use of predictive analytics in healthcare is moving slowly, but there are still issues with data accessibility. The potential of machine learning algorithms to identify pregnancies at high risk of premature delivery is demonstrated by a review of the literature. The risk of stillbirth is influenced by various factors, including mother age, BMI, and socioeconomic status. While increased mother age and BMI impede progress, improved education, easier access to prenatal care, and a decrease in maternal smoking have all contributed to a decline in stillbirth rates in the United States. Beyond current approaches, the integration of machine learning techniques could improve individual risk prediction. Risky but avoidable pregnancy problems include weight increase, blood glucose abnormalities, gestational diabetes, and abnormal blood pressure. A suggested method for identifying and forecasting these changes may be able to stop additional problems and enhance the results for both the mother and the fetus.

II. RELATED WORK

According to the author Divya Bhatnagar and Piyush Maheshwari, Cardiotocography (CTG) records fetal heart rate and uterine contractions. Their study using WEKA found J48, Random Forest, and Classification via Regression to be the best classifiers, with Random Forest excelling in high-dimensional data but being less interpretable than individual decision trees.[1]

According to the author Hoodbhoy, Zahra, Noman, Mohammad, Shafique, Ayesha, Nasim, Ali, Chowdhury, Devyani, Hasan, and Babar, obstetricians' CTG interpretations show 70% normal, 20% suspect, and 10% pathological fetal states. XGBoost, decision trees, and random forests had high precision (>96%) on training data, with XGBoost achieving the highest precision (>92%) on testing data, though it requires complex parameter tuning.[2]

According to the author M. Ramla, S. Sangeetha, and S. Nickolas, Cardiotocography records fetal heart rate and uterine contractions. They propose using CART to predict high-risk pregnancies, achieving 88.87% accuracy with entropy and 90.12% with the Gini index. While CART models are easy to interpret, they can be prone to overfitting and may require regularization or pruning.[3]

According to the author Jagannathan D, deficient interpretation of CTG can lead to unnecessary surgical interventions, like increased cesarean births. His study evaluated fetal distress using discriminant analysis (DA),

decision tree (DT), and artificial neural network (ANN), achieving accuracies of 82.1%, 86.36%, and 97.78% respectively. While these methods can capture complex nonlinear relationships and handle large datasets, they require substantial data and computational resources for training and tuning.[4]

According to the author M. Shyamala Devi, S. Sridevi, Kalyan Kumar Bonala, Ramya Harika Dadi, Kanamukkala Vinod Kumar Reddy, using metrics like Precision, Recall, F-score, Accuracy, and running time, Random Forest and Decision Tree classifiers achieved 99% and 98% accuracy respectively on a Random Oversampled dataset, and 97% and 96% accuracy using various SMOTE techniques. While Random Forest shows a slight edge in accuracy, its ensemble nature makes interpretation challenging.[5]

According to the author Yalamanchili Salini, Sachi Nandan Mohanty, and Janjhyam Venkata Naga Ramesha [6], RFs outperformed SVMs, DTs, logistic regression, k-nearest neighbors, and voting classifiers for fetal health classification using CTG data, achieving 97.5% accuracy. RFs offer feature importance estimates but may be less effective on highly imbalanced datasets.[6]

According to the author Vinayaka Nagendra Harikishan Gude, Divya Sampath, and Steven Corns [7], SVMs and Random Forests achieved over 96% accuracy in classifying fetal states (normal, suspect, pathological), with SVMs showing slight superiority for suspect cases. SVMs excel in high-dimensional spaces but require meticulous parameter tuning, involving trial and error for optimal performance.[7]

III. MATERIALS AND METHODS

A. *Dataset*

Using the cardiotocography (CTG) dataset, which tracks uterine contractions and fetal heart rate (FHR), we investigated pregnancy-related issues. The target values and crucial characteristics required for precise predictions were added to the predictive model. The dataset was separated into training and testing subsets using random sampling, yet there was still an inherent imbalance. Thirty percent of the samples were in the testing subset and seventy percent were in the training subset. This resulted in 1488 training records and 638 testing records.2126 records with CTG features were divided into three categories by three skilled obstetricians: normal, suspect, and abnormal shown in Figure 1. The machine learning model will be trained using the training data, and its performance will be evaluated using the testing data.

Following was the definition of classifications:

- Pathological: CTGs were classed as pathological if they met any of the following criteria: baseline FHR below 110 or above 160 beats per minute, decreased FHR variability, late decelerations, early decelerations, or protracted decelerations.
- Suspicious: CTGs were classed as suspicious if they did not meet both the pathological and normal criteria.
- Normal: CTGs were considered normal if they matched all of the following criteria: a baseline FHR of between 110 and 160 beats per minute, FHR variability more than 5 ms, and no late, early, or prolonged deceleration.

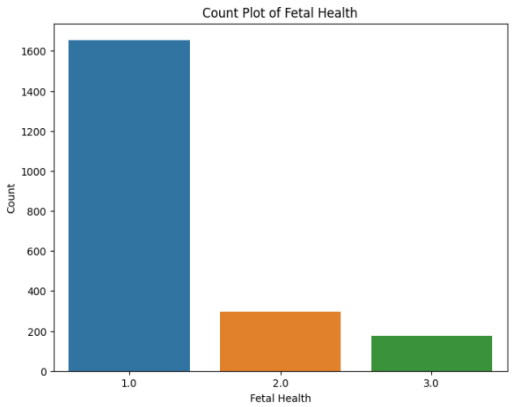


Figure 1. Unbalanced data for (i) Normal, (ii) Suspect, (iii) Pathological

B. *Models*

1) Random Forest

A random forest is a distribution with a number of decision trees of various subsets of a given feature dataset and takes the average of that dataset. The prediction accuracy increases as the number of trees grows. Forested trees provide greater consistency and eliminate overfitting problems.

Feature importance is calculated by scaling down how clean a node is, which reduces it to the probability of acquiring the node. That's the node. Price is a critical factor. For each decision tree, Scikit-learn calculates the importance of nodes using the Gini Importance by simply taking them Two children's books (bicyclic tree). All of these can then be normalized to a value from 0 to 1 by dividing by all the significant components of valuable resources. The sum of the importance of the feature in each tree is calculated and divided by the total number of trees.

$$y = \text{model}(\text{Tree}_1(X),\text{Tree}_2(X),...,\text{Tree}_{\text{Ntrees}}(X)) \quad (1)$$

where mode represents the most frequent class predicted by the trees

2)Decision Tree

Decision tree classification organizes the data into a tree-like structure where each branch represents a decision. It enables the prediction of class labels for instances by traversing the tree based on the attribute. Gini impurity: Gini impurity measures the probability of a randomly selected individual being a misclassified element of the data set.

$$1-\sum_{i=1}^{j} p_i^2 \quad (2)$$

Entropy: Entropy measures the amount of disorder or uncertainty in a data set.

$$-\sum_{i=1}^{j} p_i \log_2 (p_i) \quad (3)$$

where:

- p is $(p_1,p_2,\ldots,p_j)$ is the probability distribution of each class in a node
- j is number is classes

3) K-Neighbors Classifier

The KNN algorithm is a prediction algorithm for classification that classifies a data point based on the majority of its K nearest neighbors in the feature space. The user chooses the number of neighbors (K) as a parameter as shown in Figure 2. K in K-nearest neighbors shows the quantity of nearest neighbors considered in the making of

forecasts for a new data point. Distance is calculated by Euclidean Distance and Manhattan Distance.

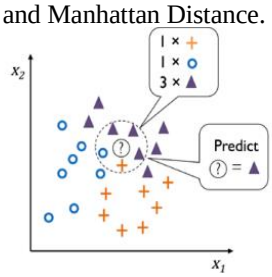


Figure 2. KNN Classifier

Euclidean Distance between two points x and y in a d-dimensional space is defined as follows:

$$\sqrt{\sum_{i=1} d(x_i - y_i)^2} \quad (4)$$

Manhattan Distance: The term referring to the distance separating two points x and y in a d-dimensional space is expressed as follows:

$$\sum_{i=1} d|x_i - y_j| \quad (5)$$

where:

- x and y are coordinates of two points.
- d is the number of dimensions.
- $|x_i - y_j|$ is absolute difference between corresponding coordinates x_i and y_j.

4) Logistic Regression

Logistic regression is a classification model that is quite easy to implement and shows great performance on step-shaped classes as shown in Figure 3. This is one of the most common classification algorithms that is used in industry.

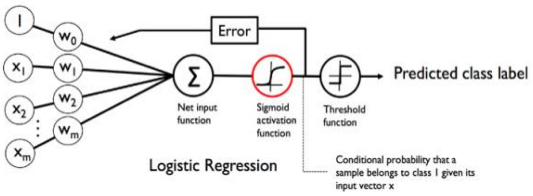


Figure 3. Flow of Logistic regression model

Input and Weights: The inputs are represented as $x_1, x_2, \ldots, x_m$. Each given input $x_i x_i$ has an corresponding weight $w_i w_i$. Furthermore, a bias term $x_0=1$ has a weight $w_0 w_0$.

Net input function: The weighted sum of the inputs is calculated by:

$$z = w_0 . x_0 + w_1 . x_1 + \ldots\ldots + w_m . x_m \quad (6)$$

Sigmoid activation function: The net input function is passed to the sigmoid activation function to obtain probability value.

Threshold function: The probability obtained from the sigmoid function is then compared to a threshold (commonly 0.5) to decide the predicted class label.

C. *Classification Metrics*

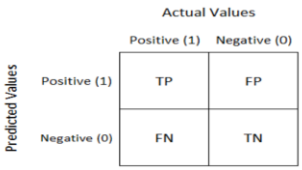
Various means by which classification performance can be measured include accuracy, confusion matrix, log-loss, and AUC-ROC, which are some of the most widely used metrics. Concerning classification issues, precision-recall serves as a commonly utilized metric.

1) Accuracy: Accuracy is a measure of how many times the predictor offers the right answer. Sometimes, accuracy can be referred to as the number of correct predictions divided by all predictions.

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

2) Confusion matrix: The confusion matrix is a table used to evaluate how well a classifier did on a data set with known outcomes.

- True Positive: In the image of a pregnant woman, the predicted result is positive, and the condition is actually true.
- True Negative: In the image of a pregnant woman, the predicted result is positive, and the condition is actually false.
- False Positive: In the image of a man suggesting a pregnant individual, the predicted result is positive, but the condition is actually false, referring to type 1 errors or false positives.
- False Negative: In the image of a woman who appears not to be pregnant, the predicted result is negative, but the condition is actually true, referring to type 2 errors or false negatives.



2) Precision: The precision of a label is described as the quotient of true positives by the number of false negatives.

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive}$$

4) Recall: It can recall a class as the portion of the correct instance that it recognizes out of the total number of positive instances.

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative}$$

5) F1 Score: The F1 score is a measure that combines precision and recall into a single value. The highest F1 score is obtained when precision is equal to recall. That is to say that the F1 score is the harmonic mean of precision and recall.

$$F1 = 2. \frac{Precision \times Recall}{Precision + Recall}$$

D. *SMOTE*

SMOTE is a technique of statistical sampling to increase uniformly the number of examples in the dataset. Based on minority cases, a new instance will then be created by the component. It takes the entire dataset as input but increases only the percentage of minority cases. Increasing by applying SMOTE doesn't alter the proportion of majority cases. The new occurrences are different from the minority cases, which already exist but are only overridden; instead, the algorithm samples from the feature space each target class and its close neighbors. After that, this sampling is used to create fresh examples, including aspects of both the target case and its neighbors as shown in Figure 4. Here, each class can use

much more characteristics, and the samples are more inclusive.

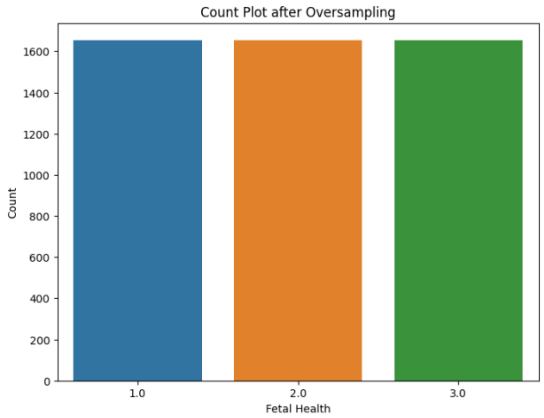


Figure 4. Balanced data for (i) Normal, (ii) Suspect, (iii) Pathological

IV. PRINCIPAL COMPONENTS ANALYSIS

Principal component analysis (PCA) is a way to reduce the number of dimensions of a dataset. In this method some of the variables are thrown away because they don't have much information content while others are retained and used as new variables. Although they are less interpretable, these are important variables because they explain the largest fraction of variance within a dataset which means they provide the most information. From a geometric standpoint, principal components constitute new coordinate axes that show the highest contrasts between observations thereby simplifying data analysis.

The number of variables in the data is the same as the principal components. The number of variables is not the same as the first principal component. The first principal component captures most of the total variance in the data set. For example, in a scatter plot, it's the line (purple) through the origin where the projection of points (red dots) is most spread out. The second principal component captures the next highest variance, and it does not correlate with the first one, and this continues until all principal components are computed. Figure 5 shows PCA for two components.

Step 1: This step is aimed at making the continuous first variables range standard so that all can have their equal share in the analysis. According to Figure, it explains the about average squared distances from projected points. Mathematically, this can be achieved by finding mean and dividing by variance on every score for all variables. Once we standardize all the variables, they will all exist on a similar scale.

$$z = \frac{value - mean}{standard\ deviation}$$

Step 2: The purpose of this particular step is to investigate the variations of the input data set variables when they are away from the mean separately. For instance, let us take a 3-dimensional data set with x, y, and z as its variables and in this case, its covariance matrix will be a 3×3 data matrix as follows:

$$\begin{bmatrix} Cov(x,x) & Cov(x,y) & Cov(x,z) \\ Cov(y,x) & Cov(y,y) & Cov(y,z) \\ Cov(z,x) & Cov(z,y) & Cov(z,z) \end{bmatrix}$$

Step 3: For simplicity's sake, let's consider a two-dimensional dataset with two variables x, y. The eigenvectors and eigenvalues of the covariance matrix are given by:

$$v1 = \begin{bmatrix} 0.6778736 \\ 0.7351785 \end{bmatrix} \qquad \lambda_1 = 1.284028$$

$$v2 = \begin{bmatrix} -0.7351785 \\ 0.6778736 \end{bmatrix} \qquad \lambda_2 = 0.04908323$$

Step 4: In the next step, following the same situations as in the last step, instead of using either of the eigenvectors v1 or v2, we can make use of both eigenvectors to form a feature vector.

$$\begin{bmatrix} 0.6778736 & -0.7351785 \\ 0.7351785 & 0.6778736 \end{bmatrix}$$

If we choose to omit the eigenvector v2, since it has less importance than its counterpart, v1 will be formed into a feature vector with that example alone.

$$\begin{bmatrix} 0.6778736 \\ 0.7351785 \end{bmatrix}$$

Step 5: Multiplying the transpose of the original data set by the transpose of the feature vector will achieve this.

$$FinalDataSet = FeatureVector^T * StandardizedOriginalDataSet^T$$

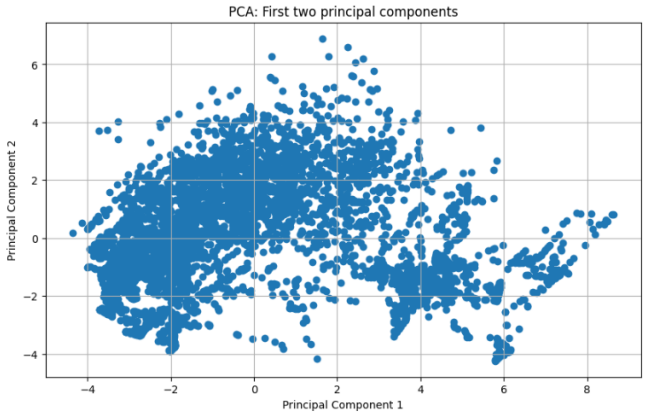


Figure 5. Principal component-1 vs Principal component-2

V. DISCUSSION

A. *Random forest*

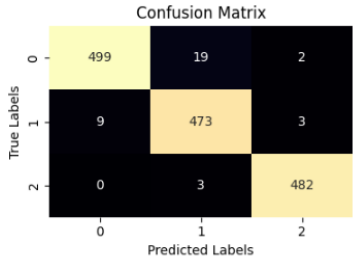


Figure 6. Confusion matrix of random forest model

Figure 6 displays the prediction of random forest model. The projected result and computational performance is displayed in the confusion matrix. The total numbers of correct predictions are 1454, with 36 incorrect forecasts.

B. *Decision tree*

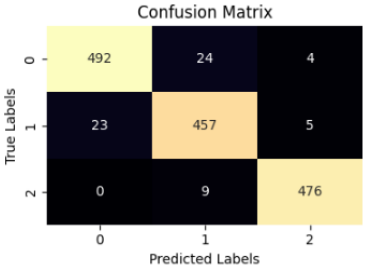


Figure 7. Confusion matrix of decision tree model

Figure 7 displays the prediction of decision tree model. The projected result and computational performance is displayed in the confusion matrix. The total numbers of correct predictions are 1425, with 85 incorrect forecasts.

Classifier Models	Class	Precision	Recall	F1 Score	Support	Accuracy (%)
Random Forest	Normal	0.98	0.96	0.97	520	
	Suspect	0.96	0.98	0.97	485	98
	Pathological	0.99	0.99	0.99	485	
Decision tree	Normal	0.96	0.95	0.95	520	
	Suspect	0.93	0.94	0.94	485	96
	Pathological	0.98	0.98	0.98	485	
K Nearest Neighbors	Normal	0.98	0.88	0.93	520	
	Suspect	0.86	0.95	0.90	485	93
	Pathological	0.95	0.96	0.96	485	
Logistic Regression	Normal	0.99	0.96	0.97	520	
	Suspect	0.94	0.98	0.96	485	97
	Pathological	0.99	0.98	0.98	485	

C. *K Nearest Neighbors*

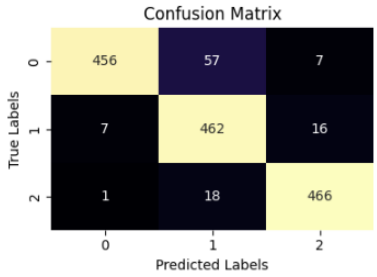


Figure 8. Confusion matrix of knn classifier model

Figure 8 displays the prediction of KNN model. The projected result and computational performance is displayed in the confusion matrix. The total numbers of correct predictions are 1384, with 106 incorrect forecasts.

D. *Logistic Regression*

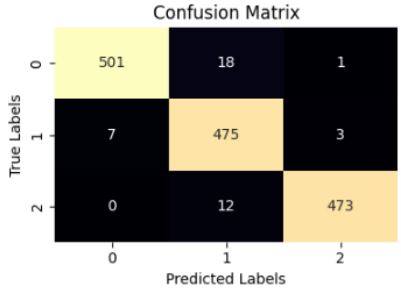


Figure 9. Confusion matrix of logistic regression model

Figure 9 displays the prediction of logistic regression model. The projected result and computational performance is displayed in the confusion matrix. The total numbers of correct predictions are 1449, with 41 incorrect forecasts.

Table I. Comparision of classifier model performance

Table I. shows the classifier model's classification report and Figure 10 shows the comparative analysis of classifier models.

VI. LIMITATIONS

a) One of the major challenges to machine learning is poor data quality and heterogeneity, which lower model performance and make it hard to generalize from noisy, inconsistent, or very diverse datasets.

b) If there's little data, it then becomes harder for the model to see patterns and make successful generalizations, reducing accuracy and dependability, hence limiting machine learning.

c) False positives and negatives misclassify the data, thus hampering the process of machine learning because of the generation of incorrect predictions that have reduced the model's effectiveness and reliability.

d) Lack of knowledge about fetal physiology and complex relationships between CTG features and fetal health decreases model accuracy and interpretability.

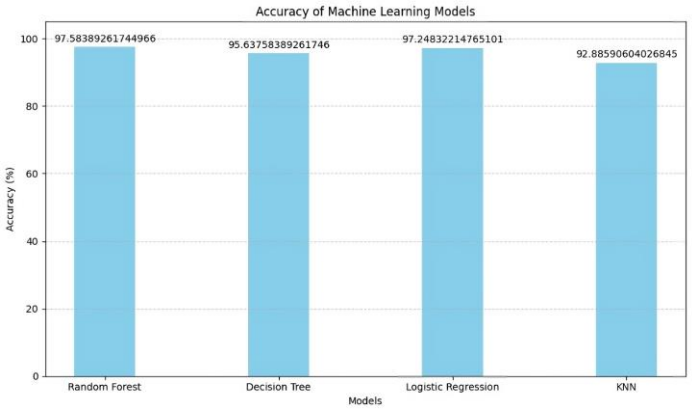


Figure 10. Comparitive analysis of classifier model

VII. CONCLUSION

In this paper, we have proposed a classification approach for fetal health data that can classify the data extracted from CTG. We have obtained the dataset from Kaggle. Our dataset contains 22 parameters. It does not contain null, duplicate, or missing values. Using this dataset, we addressed the problem of data imbalance by introducing SMOTE analysis (Synthetic Minority Oversampling Techniques). We have introduced PCA (principal component analysis) to reduce the dimensionality of parameters in the models. We have utilized five models: RF, DT, KNN and LR. We have introduced the reduced data obtained from PCA to KNN and LR. The reduced data is not efficient to operate on DT and RF since it is designed to operate on high-dimensional data. Our results demonstrated that RF outperformed the other models in terms of accuracy and f1-score. This means that, with the strength of RF, the proposed model brings better classification performance and gives better assistance.

VIII. FUTURE SCOPE

For improving maternal and fetal health outcomes, more accurate machine-learning models are required to be developed using a wide variety of maternal and fetal health data. Wearable devices and Internet of Things technologies can be integrated into maternal and fetal health to ensure continuous real-time monitoring of fetal health for timely action.

It would facilitate the realization of the delivery of insights, powered by AI and big data, and tailor-made individual plans for unique pregnancy variables to ensure personalized healthcare. That scale-up of fetal health technologies to allow affordable early detection in unserved parts of the world can greatly enhance access to essential prenatal care, bridging disparities and improving maternal and infant health outcomes globally.

IX. REFERENCES

[1] Divya Bhatnagar and Piyush Maheshwari, "Classification of cardiotocography data with WEKA", International Journal of Computer Science and Network, 2016.

[2] Zahra Hoodbhoy, Mohammad Noman, Ayesha Shafique, Ali Nasim, Devyani Chowdhury and Babar Hasan, "Use of Machine Learning Algorithms for Prediction of Fetal Risk using Cardiotocographic Data", International Journal of Applied and Basic Medical Research, Published by Wolters Kluwer - Medknow, 2019.

[3] M.Ramla, S.Sangeetha and S.Nickolas , "Fetal Health State Monitoring Using Decision Tree Classifier from Cardiotocography Measurements ", IEEE, 2018.

[4] Jagnnathan D, "Cardiotocography - A Comparative Study between Support Vector Machine and Decision Tree Algorithms", International Journal of Trend in Research and Development, 2017.

[5] M. Shyamala Devi, S. Sridevi, Kalyan Kumar Bonala, Ramya Harika Dadi, Kanamukkala Vinod Kumar Reddy. Oversampling Response Stretch based Fetal Health Prediction using Cardiotocographic Data.Annals of the Romanian Society for Cell Biology, 2021.

[6] Yalamanchili, S., Mohanty, S. N., Ramesh, J. V. N., Yang, M., &Chalapathi, M. M. V. Cardiotocography Data Analysis for Fetal Health Classification Using Machine Learning Models.International Journal of Engineering Research & Technology (IJERT), 2024.

[7] Vinayaka Nagendra Harikishan Gude Divya Sampath, Steven Corns and Suzanna Long, “Evaluation of Support Vector Machines and Random Forest Classifiers in a Real-time Fetal Monitoring System Based on Cardiotocography Data”, IEEE, 2017

