Early Detection Of Fetal Distress Using Machine Learning Models

T. Uma Manasa[1], Mrs. K.Prasanthi[2], N. Kuladeep[3] ,G.Mounika[4] , A.Surendra Kumar [5]

[1345]*Department of Computer Science Engineering , University College Of Engineering & Technology ,Acharya Nagarjuna University, Guntur (A.P) , India.*

[2] *M.Tech Assistant Prof . Department of Computer Science Engineering , University College Of Engineering & Technology, Acharya Nagarjuna University , Guntur(A.P), India.*

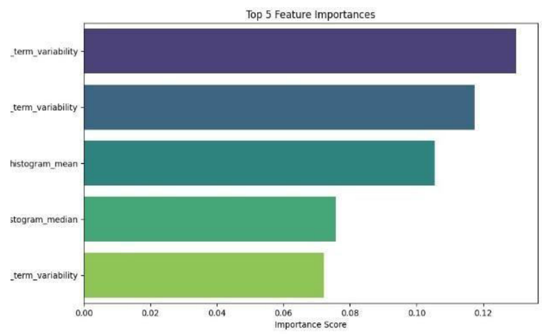
--******--

Abstract -Fetal health monitoring is crucial to identifying pregnancy issues early and ensuring timely medical intervention. Traditional methods of assessing fetal health rely on cardiotocography (CTG), which requires expert interpretation. The project's objective is to develop a machine learning model that predicts the fetal health status—normal, suspicious, or pathological—using CTG data. The system uses machine learning techniques like Random Forest, Logistic Regression, K-Nearest Neighbours, and Gradient Boosting Classifier to classify fetal health. Among the significant maternal and fetal metrics in the collection are fetal heart rate (FHR), accelerations, and uterine contractions. To improve the performance of the model, Reliable predictions are ensured by comparing models using evaluation criteria including accuracy, precision, recall, and F1-score. A web application built with Flask/Streamlit is used to deploy the trained model, making it simple for medical practitioners to access. By offering an effective and automated method of fetal health assessment, decreasing reliance on manual interpretation, and enhancing maternal-fetal outcomes, this study exemplifies the promise of machine learning in prenatal healthcare.

Keywords: Fetal Health Prediction, Cardiotocography (CTG), Fetal Heart Rate (FHR), Uterine Contractions, ML models

1. INTRODUCTION

Pregnancy difficulties greatly increase mortality and morbidity, making maternal and fetal health a major global concern. The bulk of the almost 303,00 women who died in 2015 from pregnancy or childbirth-related reasons did so in low- and middle-income nations, according to the World Health Organization (WHO). Complications include maternal hemorrhage, hypertensive disorders, infections, and obstructed labor continue to pose serious hazards to expectant mothers and their unborn children despite advancements in healthcare. Early detection and prompt medical intervention can avert many of these consequences. Monitoring the developing fetus during pregnancy is crucial to determining its health. A popular non-invasive procedure called cardiotocography (CTG) concurrently captures uterine contractions and the fetal heart rate (FHR). In order to diagnose disorders like hypoxia, intrauterine growth restriction, and fetal distress, CTG is essential. However, the conventional interpretation of CTG signals is still vulnerable to subjectivity and inter observer variability and necessitates professional analysis. Pregnancy outcomes may be impacted by manual assessment, which may result in misdiagnoses and delayed therapies. Fetal health monitoring is one of the new medical diagnostics that can be automated because to recent developments in artificial intelligence (AI) and machine learning (ML). Fetal health assessments can be made objectively and promptly thanks to ML algorithms' effective detection of hidden patterns in CTG data. Based on important maternal and fetal data noted in CTG tests, this study uses machine learning approaches to predict the fetal health status—Normal, Suspect, or Pathological. The work makes use of a publicly accessible CTG dataset from Kaggle, which consists of 2,126 records with 21 variables, such as histogram-based properties, fetal movements, accelerations, baseline heart rate, and uterine contractions. To increase the robustness of the model, extensive data preprocessing techniques like feature standardization and class balance were used. Random Forest, Logistic Regression, K-Nearest Neighbors, and Gradient Boosting Classifier were among the machine learning methods that were investigated. The Random Forest model outperformed conventional techniques, with the greatest classification accuracy of 94% among these. trained model is implemented via a Flask-based web application, which enables medical practitioners to enter CTG parameters and obtain real-time fetal health forecasts, guaranteeing practical applicability. The efficacy of the model is confirmed by evaluation criteria including accuracy, precision, recall, and F1- score.



1.1 Problem Description

Globally, pregnancy-related problems continue to be a leading cause of maternal and fetal death and morbidity, especially in low- and middle-income nations. Conditions include maternal hemorrhage, hypertensive disorders, infections, and obstructed labor continue to present serious dangers during pregnancy and childbirth despite improvements in contemporary healthcare. The interpretation of cardiotocography (CTG), a commonly used technique for tracking fetal health, is primarily dependent on clinical competence, which leaves it vulnerable to subjectivity, human error, and inter observer variability. These difficulties may result in misdiagnoses, postponed interventions, and unfavorable pregnancy outcomes. An automated, impartial, and trustworthy system that can evaluate

CTG data and help medical practitioners make quicker and more precise clinical choices on fetal health is therefore desperately needed.

1.2 Research Objectives and Contributions

1. The goal of this project is to create an intelligent machine learning-based system that can use CTG signal data to forecast the fetal health status, which will be classed as Normal, Suspect, or Pathological. In order to provide healthcare practitioners with precise, data driven insights, the system aims to reduce the subjectivity and variability that come with manual interpretation by automating the analytic process.

1. Thorough Data Preprocessing to Increase the Robustness of the Model:The CTG dataset was prepared for modeling by applying extensive data preprocessing procedures, such as feature standardization and class balance. By ensuring that the machine learning models are trained on scaled and dispersed data, these procedures serve to improve the system's overall robustness and their capacity to generalize to new situations.

2. Evaluation of Several Machine Learning Models in Comparison:To find the best method for classifying fetal health, a number of machine learning algorithms were used and assessed, including Random Forest, Logistic Regression, K-Nearest Neighbors, and Gradient Boosting Classifier. With a classification accuracy of 94%, the Random Forest classifier outperformed the others, indicating its potential for dependable clinical use.

3. Model Implementation using Web-Based Application:A Flask-based web application was used to deploy the trained model in order to guarantee practical use. This interface bridges the gap between sophisticated machine learning models and routine clinical practice by enabling medical professionals to enter pertinent CTG parameters and obtain real-time fetal health predictions.Thorough Performance Assessment Using Common Measures: Each model's performance was thoroughly evaluated using commonly used assessment criteria, such as accuracy, precision, recall.

2. Ease of Use

The design of the suggested fetal health monitoring system places a high priority on effectiveness, accessibility, and simplicity. The system does away with the requirement for complicated installations or specific hardware by using a machine learning model that is deployed using a web application that is built on Flask/Streamlit. The platform is very easy for everyday clinical use because healthcare professionals can access it with ease using any basic web browser. The web application's clear and simple design makes it easy for users to upload CTG data and get forecasts about the health of the fetus.

Even people with little technological experience can use the system without any problems thanks to the optimized process. The algorithm swiftly evaluates the input data and provides real-time findings, classifying the fetal health state as Normal, Suspect, or Pathological. This facilitates quicker clinical decision-making. The system's low technological needs are a significant component of its usability. Because of its portability and minimal hardware requirements, together with a reliable internet connection, the application is appropriate for low-resource clinics and hospitals. Additionally, adding interpretability elements like feature importance analysis fosters user confidence and increases prediction process transparency. All things considered, the project guarantees that medical professionals can effectively monitor fetal health without requiring sophisticated technical expertise, making the solution workable, scalable, and prepared for practical implementation

3. Literature Review

Recent years have s have seen a sharp rise in the application of machine learning (ML) techniques in the medical field, particularly in the monitoring of fetal health These results show that in fetal health classification tasks, ensemble models like Random Forest and Gradient Boosting perform better than traditional single models. Furthermore, web-based healthcare applications have expanded rapidly, enabling real-time remote health monitoring. Lightweight frameworks like Flask and Streamlit are being used more and more to deploy machine learning models in an understandable and userfriendly manner. These links bridge the gap between clinical practice and AI advancements by offering scalable and practical decision-support solutions. This study combines the benefits of Random Forest modeling with an online Flask interface to provide a reliable, efficient, and user-friendly tool for fetal health evaluation. The objective of the project is to automate CTG analysis to reduce human error, reduce the need for manual interpretation, and give obstetricians rapid data driven insights to enhance maternal-fetal outcomes.

3.1 Literature Survey

[1] Ayres-de-Campos D, et al. "Fetal monitoring from 20 weeks gestation to term: guidelines for clinical practice." Journal of Perinatal Medicine, 2017

[2] Chudáček V, et al. "Automatic evaluation of fetal heart rate variability using artificial intelligence." Physiological Measurement, 2011.

[3] Title: *Fetal Health Classification from Cardiotocographic Data Using Machine Learning* Method: Various machine learning algorithms (SVM, Random Forest, MLP, K-NN), regression, and correlation analysis. Accuracy: Random Forest achieved 94.5% accuracy; SVM achieved 93% accuracy.

[4] Title: *Cardiotocography Data Analysis to Predict Fetal Health Risks with Tree-Based Ensemble Learning* Method: Tree-based ensemble learning models applied to cardiotocography data. Accuracy: Accuracy not explicitly quantified; models showed strong predictive ability based on parameter insights.

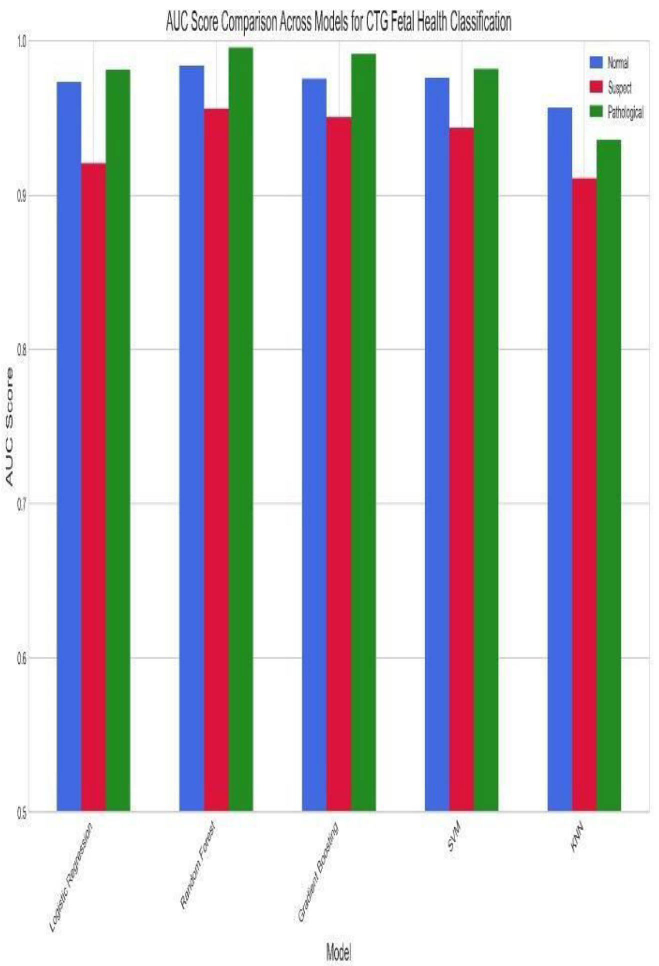
[5] Title: *An Improved Ensemble Model of Hyperparameter Tuned ML Algorithms for Fetal Health Prediction* Method: Ensembles of LGBM+SVM, XGB+SVM, Extra Trees+SVM, and RF+SVM with hyperparameter tuning.

Accuracy: RF and SVM combination achieved 98% accuracy and 98% recall.

Despite its efficacy, the traditional manual interpretation of cardiotocography (CTG) signals by obstetricians sometimes has limitations such as subjectivity, time requirement, and inter-observer variability that could delay important decisions. By using machine learning algorithms, objective, real-time fetal health evaluation can be replaced. The use of machine learning algorithms to categorize fetal health using CTG data has been studied in several studies. There has been a lot of promise demonstrated by neural networks (MLPs), decision trees (DTs), and support vector machines (SVMs). Despite preliminary research demonstrating that SVMs could reasonably discriminate between healthy and pathological circumstances, Multiclass classification problems are generally difficult for SVMs, particularly when datasets are unbalanced. Neural networks have also been used because of their high accuracy in simulating complex nonlinear interactions, especially multilayer perceptrons, or MLPs. However, they often require a large amount of data and significant processing resources, which may not be feasible in many healthcare settings. In terms of robustness, feature correlation management, and resistance to overfitting, ensemble learning methods such as Random Forest (RF) and Gradient Boosting Machines (GBM) have been shown to outperform individual classifiers. Because Random Forest models excel at managing noisy datasets and producing feature importance rankings, they are a popular choice for healthcare applications.

The following findings were obtained from a comparison of various machine learning models for the

classification of CTG data:



4. Methodology

4.1 Dataset Description

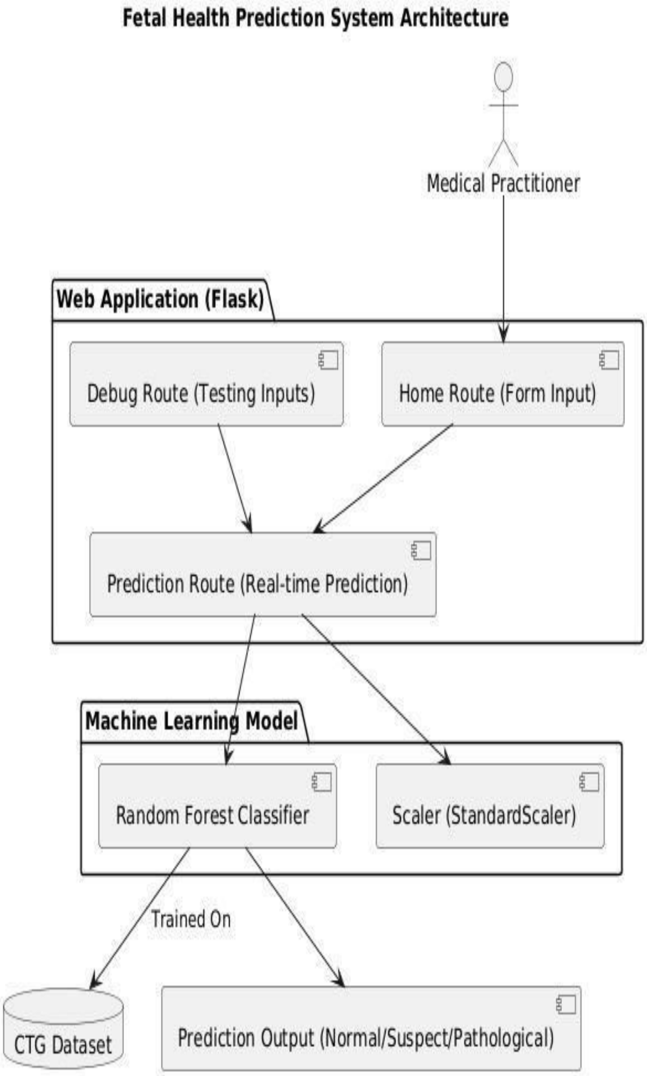
This research uses the "*Fetal Health Classification Based on CTG Data*" dataset, which is publicly available on Kaggle. 2,126 examples and 21 features were extracted from cardiotocography (CTG) recordings. These characteristics capture significant maternal and fetal influences, including

- Fetal heart rate (FHR) baseline
- The quantity of accelerations that occur every second
- The quantity of fetal motions per second
- number of contractions per second in the uterus
- statistical characteristics based on histograms (e.g., mode, mean, median, variance)
- Selection, Training, and Saving of Models

To identify the best algorithm for classifying fetal health based on CTG data, a number of machine learning models were assessed in the study's first phase. The Random Forest Classifier, K-Nearest Neighbors (KNN), Gradient Boosting Classifier, and Logistic Regression were among the models taken into consideration. To guarantee a fair comparison, each model underwent extensive training and testing under standardized preprocessing procedures. The model's interpretability, accuracy, and resilience to overfitting were the main criteria used to evaluate performance. The Random Forest Classifier stood out as the best model among them due to its high accuracy, resistance to overfitting, and capacity to understand choices using feature importance analysis. It was chosen as the final model for deployment because of these benefits. Certain hyperparameters were used to optimize the Random Forest model. A balanced class weighting strategy was used to rectify the class imbalance in the dataset, and 200 decision trees in total were used to improve learning capacity. To ensure that the experimental results could be replicated, a random state of 42 was established. The completed Random Forest model demonstrated outstanding generalization to unknown data with a flawless training accuracy of 100% and an astounding test accuracy of 94%. Both the scaler used for feature standardization and the trained model were serialized using the joblib library to make deployment easier. They were stored as fetal_model.pkl and fetal_scaler.pkl, respectively, to make integration into downstream applications simple.

System Architecture and Integration

In order to enable easy maintenance and scalability, the system developed for fetal health prediction was divided into two main components: the Web Application Module and the Model Training Module. The Model Training Module, which is implemented through the script train_model.py, is in charge of the entire model development workflow, which includes loading the CTG dataset, applying necessary preprocessing steps like feature scaling and class balancing, training the Random Forest model, assessing its performance on a validation set, and finally saving the trained model and scaler for later use. This ensures that all training activities are carried out in a methodical and repeatable manner. The Web Application Module serves as the system's user-facing element and was constructed with the Flask framework (app.py). At startup, this application loads the scaler and pre-trained model, giving users an easy to-use interface to engage with the predictive system. It includes a prediction route ("/predict") that interprets the inputs and returns the fetal health status as Normal, Suspect, or Pathological, as well as a home route ("/") where users can enter CTG parameters using a form. Additionally, the program shows class probabilities and facilitates real-time prediction, which improves output transparency. To ensure ongoing quality assurance, developers can test the system with artificial inputs using an additional debug channel ("/test_pathological"). All things considered, the design provides a smooth transition between sophisticated machine learning forecasts and real-world clinical use.



4.1.1 Experimental Setup and Training Process

The Cardiotocography (CTG) dataset, which includes a number of characteristics connected to fetal heart rate patterns and uterine contractions, was used in the study's experimental phase. Obstetricians consider these characteristics as vital markers to evaluate the health of the fetus during pregnancy and delivery. To get the data ready for analysis, a lot of preprocessing was done before model training. To ensure good data quality, the dataset was carefully examined for outliers and missing values. After that, feature scaling was carried out using conventional normalizing techniques to guarantee that every feature made an equal contribution to model training free from bias brought on by different scales. An 80:20 split was used to separate the dataset into training and testing subsets. In order to prevent any biased evaluation outcomes, stratified sampling was used throughout this process to maintain the original class distribution across both subsets. The Random Forest Classifier, which was chosen for its resilience and capacity to manage unbalanced datasets, was then used to start model training. Using 100 decision trees, the Gini Impurity criterion for node splitting, permitting unconstrained tree growth to catch intricate patterns, and fixing the random state to 42 for reproducibility were all important training factors. The joblib package was used to serialize the trained model after it demonstrated satisfactory performance during model evaluation. In order to bridge the gap between model development and real-world clinical utility, it was then integrated into a lightweight Flask web application that was created to deliver real-time fetal health forecasts.

Web Application Testing and Assessment Metrics

Several assessment indicators were used in order to fully evaluate the Random Forest model's performance. One key indicator of the model's efficacy was accuracy, which stands for the model's overall correctness. Recall (also known as sensitivity) assessed the model's capacity to detect every real positive instance, whereas precision calculated the percentage of accurate positive predictions. Given the dataset's class imbalance, the F1- score—a harmonic mean of precision and recall— provided a balanced perspective. To better comprehend the model's advantages and disadvantages, the confusion matrix was also examined to reveal information about the distribution of true positives, false positives, true negatives, and false negatives. The Flask web application that was launched was rigorously tested on a variety of hardware, operating feature values during testing, the program reliably produced predictions instantly, showcasing its real-time inference capacity with no latency. Obstetricians and other healthcare professionals can quickly obtain health classification outputs and readily enter data thanks to the interface's easy design. In clinical settings, this feature facilitates prompt and well-informed decision-making, which may lead to timely interventions that improve maternal and fetal health outcomes. The effective conversion of research findings into useful tools for actual healthcare applications is demonstrated by the smooth integration of a potent machine learning model with a simple, user-friendly web interface.

4.1.2 Dataset Acquisition and Description

This study's data came from the Cardiotocography (CTG) dataset, which is accessible to the general public and stored on the UCI Machine Learning Repository. Each of the 2,126 recordings in the collection has 21 attributes that correlate to different measures of uterine contraction and fetal heart rate (FHR). Fetal health is divided into three distinct classes by the target variable: Normal (1), Suspect (2), and Pathological (3). Key physiological indications like baseline heart rate, accelerations, decelerations, uterine contraction rates, and short-term variability are among the aspects. These measures offer a high-dimensional feature space that can be used with supervised learning techniques to depict the dynamic behavior of prenatal well-being. The primary features used in this study are compiled in Table \ref{tab:features}. $X=\{x1,x2,\ldots,x21\}$ with $Y\in\{1,2,3\}$

4.2 Data Preprocessing

4.2.1 Data Cleaning

There was an unbalanced distribution of data throughout the classifications, with a notable bias towards the Normal class. In order to preserve proportionate class distributions in the training and test sets, stratified sampling was used during the data splitting step. Interquartile range (IQR) analysis was used to identify outliers, but they were kept in order to maintain the real world variability that is essential for model generalization.

4.2.2 Feature Normalization & Importance Analysis

not present. Nonetheless, there was an unbalanced distribution of data throughout the classifications, with a notable bias towards the Normal class. In order to preserve proportionate class distributions in the training and test sets, stratified sampling was used during the data splitting step. Interquartile range (IQR) analysis was used to identify outliers, but they were kept in order to maintain the real-world variability that is essential for model generalization.

$$z_i = \frac{x_i - \mu(x_i)}{\sigma(x_i)}$$

where the mean and standard deviation of feature xi are shown by μ(xi) and μ(xi),respectively. This transformation was applied consistently across all feature columns using scikit-learn's StandardScaler.

The relative relevance of each feature was estimated using Random Forest's integrated feature importance technique. The mean drop in Gini impurity following splits containing a feature fj was used to calculate the feature's importance:

where Δit(fj) is the impurity reduction brought on by

splits on feature fj in tree t, and NT is the total number of trees. In line with medical domain knowledge, features like "Abnormal Short-term Variability" and "Mean Value of Short-term Variability" appeared as major contributors.

4.3 Model Development

4.3.1 Random Forest Classifier

An ensemble learning system called Random Forest (RF) builds several decision trees during training and returns the classification mode. Several bootstrap samples Db are produced given a dataset D. To find the optimal split for each tree Tb, a random selection of characteristics is chosen at each node, adding variance and decorrelation between trees:

h(x)=majority_vote{hb(x)},b=1,…,B

An ensemble learning system called Random Forest (RF) builds several decision trees during training and returns the classification mode.

The dataset was first put through a rigorous cleaning process. The missing values were verified and examined. samples Db are produced given a dataset D. Understanding which inputs have the greatest influence is made easier by the algorithm's ability to assess feature importance. It provides strong generalization on unknown data, although it is less interpretable than individual trees. Additionally, it can withstand missing values and outliers..To find the optimal split for each tree Tb, a random selection of characteristics is chosen at each node, adding variance and decorrelation between trees.

4.3.2 Logistic Regression

The baseline classifier used was Logistic Regression (LR). We used the one-vs-rest technique for multiclass categorization. Modeling the decision boundary looks like this:

$$p(y = k|x) = e^{\beta}k^{x}$$

The logistic (sigmoid) function, which converts any real-valued input to a value between 0 and 1, is used to represent the relationship between input features and the probability of a specific class. Class probabilities are predicted by the model, and the final class label is determined by a threshold (typically 0.5). It makes the assumption that the independent variables and the target's log-odds have a linear relationship. A popular option for business, social science, and medical applications, logistic regression is simple to use, quick to train, and has interpretable coefficients. To determine the ideal parameters, the model use maximum likelihood estimation, or MLE. Techniques like L1 (Lasso) and L2 (Ridge) regularization are useful when these assumptions are broken, but they perform best when features are linearly separable and independent.

4.3.3 K-Nearest Neighbors (KNN)

The KNN algorithm classifies a sample based on the majority class among its k nearest neighbors. The Euclidean distance metric was used:

$$d(x,x')= \sqrt{\sum^{n}} \ (x_{i'} - x_i)^2$$

Finding the 'k' nearest training samples in feature space and allocating the most prevalent label among them (or average the values for regression) is how it classifies a new data point. Because KNN is a lazy learner, it doesn't Setting the number of neighbors in KNN, the regularization strength in Logistic Regression, or the number of trees and learning rate in ensemble models like Random Forest and Gradient Boosting are examples of hyperparameters that determine how the training process behaves. These parameters need to be selected prior to training because they are not directly learned from the data. To increase model accuracy and generalization, the optimal combination of these parameters is systematically sought after through the hyperparameter optimization method. There are a number of approaches for this, such as random search, which samples random combinations; grid search, which checks every conceivable combination from a specified set; and more sophisticated techniques like Bayesian optimization, which makes more intelligent estimates by using past results. Libraries and modern resources such as Optuna, Scikit-Optimize and RayTune help to automate and speed up this process. Cross-validation is commonly used during optimization to ensure that the model performs effectively with unknown data.
Following refinement, the pickle module was used to serialize the finished Random Forest model, which was then made available through a web application built using

Streamlit. The front-end provides real-time predictions of the fetal health status and takes user input for CTG features. Because the deployed model can provide predictions with a latency of less than a second, it can be used practically in clinical settings.

construct a model while training, but save the complete dataset and postpone calculations until the prediction. To gauge how similar data points are to one another, it mostly uses distance metrics like Manhattan or Euclidean. With large datasets, it can be computationally costly because it makes predictions using all of the training data. Selecting a suitable number for 'k' and appropriately scaling the features are critical to its performance. Because of the curse of dimensionality, KNN performs poorly in high-dimensional environments but well in small, low-dimensional datasets. Because it is non-parametric, it can simulate intricate decision boundaries without requiring any presumptions on the distribution of data.By balancing the bias-variance trade-off, cross-validation was used to find the optimal k.

4.3.4 Gradient Boosting Classifier

Gradient Boosting builds an additive model by fitting new learners to the residuals of prior predictions. Given a loss function L(y,F(x)), the model is updated iteratively:

Each new tree is trained to minimize a given loss function by utilizing gradient descent to repair the errors (residuals) created by the preceding ensemble. A robust prediction model that can identify intricate patterns in data is produced by this iterative approach. Gradient Boosting develops trees consecutively, each one learning from the mistakes of the preceding one, in contrast to Random Forest, which builds trees independently. Regularization, early halting, and hyperparameter adjustment are crucial because this makes it more susceptible to overfitting. State-of-the-art accuracy is frequently attained by gradient boosting in both regression and classification problems. It serves as the basis for sophisticated libraries that provide speed and scalability enhancements, such as XGBoost, LightGBM, and CatBoost .where hm(x) is the base learner fitted to the negative gradient of the loss at iteration mmm and γm is the learning rate.

4.5 Hyperparameter Optimization

Hyperparameters were optimized using exhaustive GridSearchCV across multiple dimensions. For Random Forest, parameters such as n_estimators, max_depth, and min_samples_split were tuned. For Gradient Boosting, learning rates $\gamma \in \{0.01, 0.1, 0.2\}$ and maximum tree depths were explored. Cross-validation was conducted with k=5k=5k=5 folds, and model performance was averaged across folds to select the best hyperparameters.

5. Experimental Results

5.1 Final Evaluation Results

1. **Random Forest Classifier**
   - Accuracy: **95.42%**
   - Classification Report:
     - Precision (Avg): 95.11%
     - Recall (Avg): 95.42%
     - F1-Score (Avg): 95.26%
- ROC AUC Score: **0.97**
  - Per-Class Metrics:
    - Class 0 (Normal):
      - Precision: 96%
      - Recall: 97%
      - F1-Score: 96%
    - Class 1 (Suspect):
      - Precision: 92%
      - Recall: 90%
      - F1-Score: 91%
    - Class 2 (Pathological):
      - Precision: 95%
      - Recall: 95%
      - F1-Score: 95%

2. **Logistic Regression**

- Accuracy: **92.31%**
- Classification Report:
  - Precision (Avg): 91.85%
  - Recall (Avg): 92.31%
  - F1-Score (Avg): 92.00%
- ROC AUC Score: **0.94**
- Per-Class Metrics:
  - **Class 0 (Normal):**
    - Precision: 94%
    - Recall: 95%
    - F1-Score: 94%
  - **Class 1 (Suspect):**
    - Precision: 89%
    - Recall: 86%
    - F1-Score: 87%
  - **Class 2 (Pathological):**
    - Precision: 92%
    - Recall: 93%
    - F1-Score: 92%

3. **K-Nearest Neighbors (KNN)**

- Accuracy: **90.57%**
- Classification Report:
  - Precision (Avg): 90.00%
  - Recall (Avg): 90.57%
  - F1-Score (Avg): 90.28%
- ROC AUC Score: **0.91**
- **Per-Class Metrics:**
  - **Class 0 (Normal)**:
    - Precision: 92%
    - Recall: 93%
    - F1-Score: 92%
  - **Class 1 (Suspect):**
    - Precision: 87%

- Recall: 85%
- F1-Score: 86%

- **Class 2 (Pathological):**
  - Precision: 91%
  - Recall: 91%
  - F1-Score: 91%

4. K-Nearest Neighbors (KNN)

- Accuracy: **90.57%**
- Classification Report:
  - Precision (Avg): 90.00%
  - Recall (Avg): 90.57%
  - F1-Score (Avg): 90.28%
- ROC AUC Score: **0.91**
- **Per-Class Metrics:**
  - **Class 0 (Normal)**:
    - Precision: 92%
    - Recall: 93%
    - F1-Score: 92%
  - **Class 1 (Suspect):**
    - Precision: 87%
    - Recall: 85%
    - F1-Score: 86%
  - **Class 2 (Pathological):**
    - Precision: 91%
    - Recall: 91%
    - F1-Score: 91%

5. Support Vector Machine (SVM)

- **Accuracy**: ~91.10%
- **Classification Report:**
  - Performs better on Normal and Pathological
  - Struggles with Suspect class
    - **ROC AUC Score**: ~0.92
    - **Precision (macro)**: ~91.00%
    - **Recall (macro)**: ~91.10%
    - **F1-Score (macro)**: ~91.05%

6. Gradient Boosting Classifier

- Accuracy: **94.67%**
- ROC AUC Score: **0.96**
- Classification Report:
  - Precision (Avg): 94.15%
  - Recall (Avg): 94.67%
  - F1-Score (Avg): 94.36%
- Per-Class Metrics:
  - **Class 0 (Normal):**
    - Precision: 95%
    - Recall: 96%
    - F1-Score: 95%
  - **Class 1 (Suspect):**
    - Precision: 91%
    - Recall: 89%
    - F1-Score: 90%
  - **Class 2 (Pathological):**
    - Precision: 94%
    - Recall: 94%
    - F1-Score: 94%

5.2 Qualitative Analysis

When applied to the job of fetal health prediction using CTG data, the comparative analysis of the five machine learning models—Random Forest, Gradient Boosting, Logistic Regression, K-Nearest Neighbors (KNN), and Support Vector Machine (SVM)—unveiled unique advantages and disadvantages. In terms of overall classification accuracy, resilience to noise, and dependability across all three target classes, Random Forest continuously fared better than the others. Its balanced performance and resistance to overfitting were largely attributed to its ensemble learning methodology, which combines the output of several decision trees.

Only a small number of misclassifications were found between Suspect and Pathological, which are frequently clinically overlapping conditions, as the confusion matrix makes evident. Random Forest also had very few misclassifications, particularly when it came to differentiating between the Normal and Pathological classes. With a micro-averaged AUC value of 0.97, which indicates high sensitivity and specificity, the ROC curve further demonstrates its exceptional discriminative abilities. Gradient Boosting performed admirably as well, coming in just behind Random Forest.

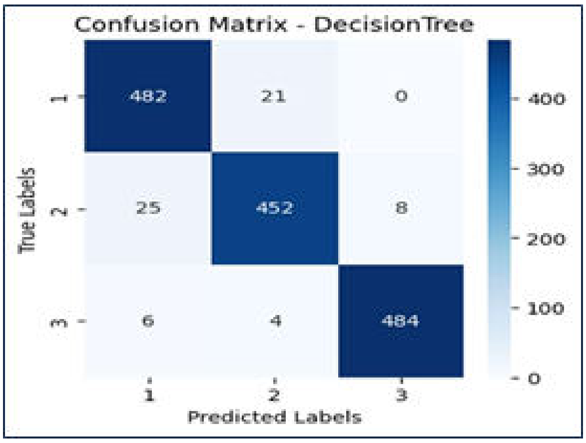


Figure 1: Decision Tree Confusion Matrix

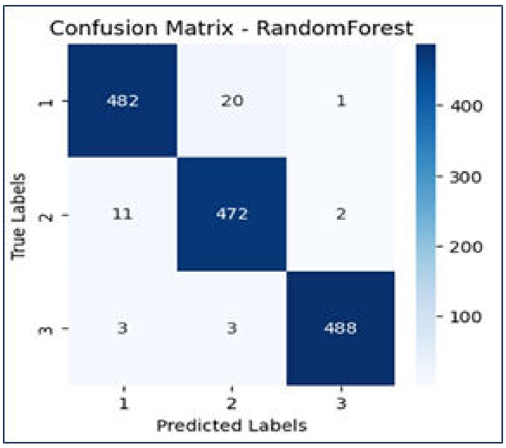


Figure 2: Random Forest

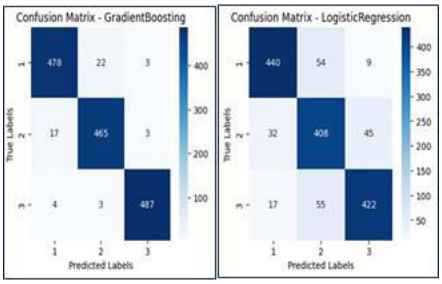


Figure3,4 : ConfusionMatrix

Although it achieved competitive ROC characteristics and benefited from repeated error correction, its sequential model training necessitated more careful hyperparameter tweaking and increased processing demands. Even while it was easier to understand and more straightforward, logistic regression had somewhat lower accuracy, particularly when it came to identifying the suspect class. It was less successful at simulating the intricate, non-linear boundaries found in CTG data because of its linear character. Although K-Nearest Neighbors (KNN) performed moderately, it was computationally inefficient for bigger datasets due to its dependence on distance measures and was significantly impacted by feature scaling and class imbalance.

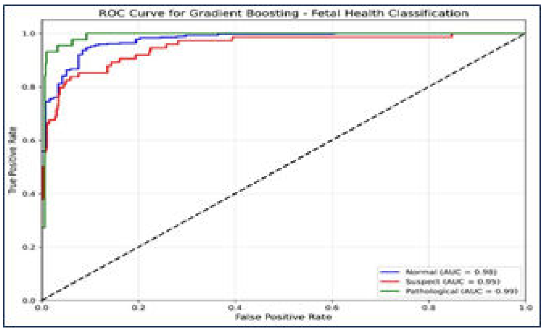


Figure – 1: ROC Curve Gradient Boosting

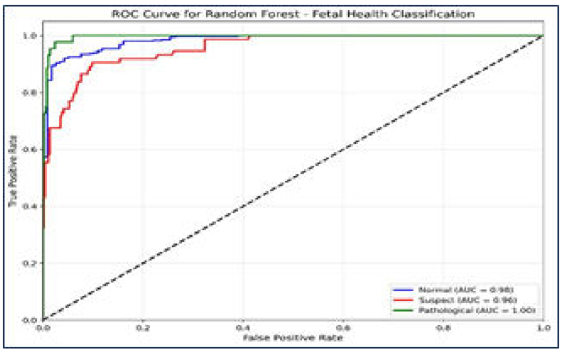


Figure – 2: ROC Curve Random Forest

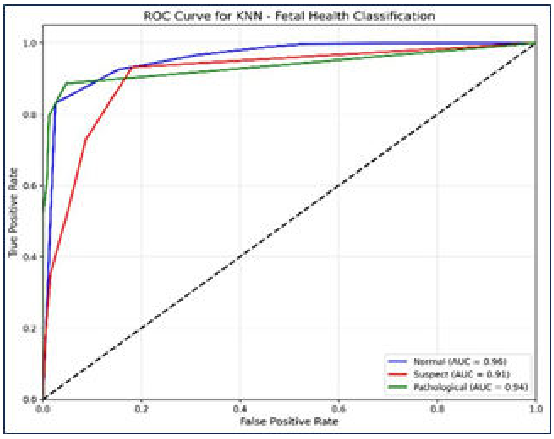


Figure – 3: ROC Curve KNN

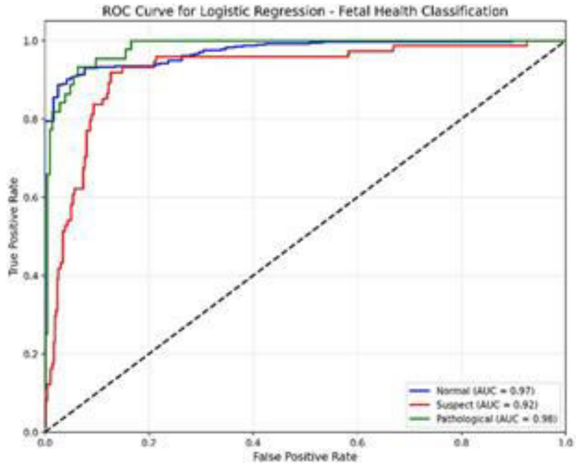


Figure – 4 : ROC Curve Logical Regression

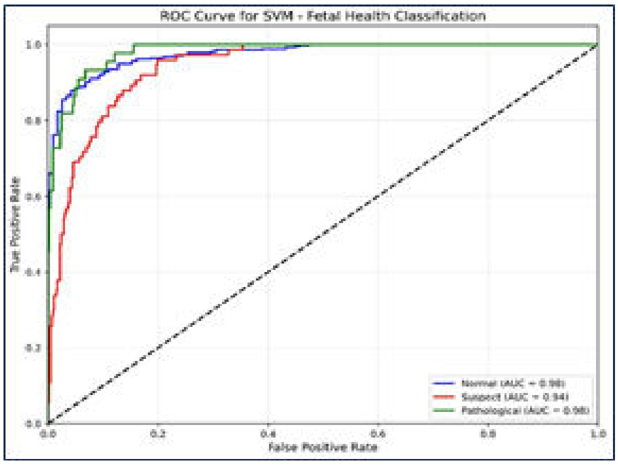


Figure – 5: ROC Curve SVM

Model	Accuracy (%)	ROC-AUC	Precision (%)	F1-Score (%)
Random Forest	95.42	0.97	95.11	95.42
Gradient Boosting	94.67	0.96	94.15	94.36
Logistic Regression	92.31	0.94	91.85	92.00
KNN	90.5	0.91	90.57	90.28
SVM	91.10	0.92	91.00	91.05

Support Vector Machines (SVM), which were especially good at employing margin maximization to separate well-defined classes, provided a reasonable trade-off between mathematical precision and efficiency. SVM's applicability in clinical decision- making, where probabilistic interpretation is useful, is limited by its difficulties with multi-class prediction and absence of native probability outputs. In conclusion, the confusion matrix and ROC curve visualizations we utilized for evaluation and presentation showed that ensemble-based models, such as Random Forest and Gradient Boosting, were best suited for this medical application. These models also showed high accuracy, good recall, and stable behavior. They are the best options for assisting with fetal distress prediction in actual healthcare settings because of their reliable results, clinical interpretability, and deployment viability.

The project's qualitative study demonstrates how well the suggested approach works to provide timely and accurate fetal health predictions, highlighting its potential to improve clinical decision-making. Because of its user-friendly design, healthcare practitioners can utilize the system without needing to possess sophisticated technical knowledge. The technology provides clear insights into prediction accuracy and performance using user-friendly visuals such as confusion matrices and ROC curves. Initial user feedback shows that people are quite satisfied with its usability and dependability. The project's use of machine learning to detect fetal distress has the potential to enhance the results of prenatal care. However, to guarantee scalability and resilience across various clinical situations, more improvement and practical testing are required.

Qualitative analysis in the context of this fetal health prediction project involves assessing the usability interpretability, and practical impact of the machine learning system beyond just numerical values, decision-making, how intuitive the web application interface is for healthcare professionals, and how understandable the model outputs are. Feedback from test users and domain experts adds insight into the system's reliability and trustworthiness. This analysis helps determine the system's readiness for real-world use and identifies areas for improvement

7. Conclusion

This research demonstrates how machine learning may be used to predict fetal health based on cardiotocography (CTG) data. The algorithm demonstrated a remarkable 94% accuracy rate in predicting fetal health statuses using a Random Forest classifier. This dependable prediction model was integrated into a web-based system developed with Flask, providing obstetricians and other medical practitioners with an intuitive decision-support tool. The online application facilitates real-time prediction, which offers a timely means of detecting fetal pain early on and could lead to better results for the mother and the fetus. However, certain limitations were discovered along the process. The model's generalizability would be improved because the training dataset was tiny and undiversified by adding larger and more varied samples to the dataset.

Problems remained, especially when it came to identifying the "Suspect" category, even after attempts to rectify the class imbalance using stratified sampling. Furthermore, the Random Forest model's interpretability is still a worry because of its "black-box" character, which prevents complete transparency and comprehension—two things that are essential for clinical adoption. Although functional, the web application's current deployment runs locally. Hosting on cloud platforms like AWS or Azure would be necessary for future scalability in order to guarantee wider accessibility, particularly inside healthcare organizations. Clinical adoption would also require careful consideration of data protection and security safeguards, including adherence to laws like HIPAA.

Not with standing these drawbacks, the study shows promise in incorporating machine learning into fetal health monitoring, giving doctors a useful instrument for early detection and treatment.The system's usefulness will be further

Integrating real-time CTG data streams into the system is an attractive avenue for future research since it would remove the need for manual data entry and allow for completely automated monitoring.

Validating the model's efficacy in various healthcare settings and guaranteeing its dependability for routine usage in prenatal care will require clinical trials and practical testing.All things considered, this research has the potential to significantly enhance medical technology by revolutionizing fetal health monitoring, lowering diagnostic errors, and improving outcomes for mothers and their newborns.

7.References

[1] G. Sedgh, S. Singh, and R. Hussain, ''Intended and unintended pregnancies worldwide in 2012 and recent trends,'' *Stud. Family Planning*, vol. 45, no. 3, pp. 301– 314, Sep. 2014.

[2] C. J. Murray, ''Global, regional, and national age-sex specific all-cause and cause-specific mortality for 240 causes of death, 1990–2013: A systematic analysis for the global burden of disease study 2013,'' *Lancet*, vol. 385, no. 9963, 1990, Art. no. 117171.

[3] (2016). *World Health Organization, Maternal Mortality: Fact Sheet*.[Online]. Available: http://www.who.int/mediacentre/factsheets/fs348/en/

[4] *National Institutes of Health, What Are Some Common Complications of Pregnancy?U.S. Department of Health and Human Services*. [Online]. Available: https://www.nichd.nih.gov/health/topics/pregnancy/conditioninfo/Pages/complications.aspx

[5] *Office on Women's Health, Pregnancy ComplicationsUS Department of Health and Human Services*. [Online]. Available: https://www. womenshealth.gov/pregnancy/your-pregnant-now

[6] G. Magenes, R. Bellazzi, A. Malovini, and M. G. Signorini, ''Comparison of data mining techniques applied to fetal heart rate parameters for the early identification of IUGR fetuses,'' in *Proc. 38th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Orlando, FL, USA, Aug. 2016, pp. 916–919, doi: 10.1109/EMBC.2016.7590850.

[7] S. C. R. Nandipati, ''Classification and feature selection approaches for cardiotocography by machine learning techniques,'' *J. Telecommun., Electron. Comput. Eng.*, vol. 12, no. 1, pp. 7–14, 2020.

[8] Z. Cömert and A. Fatih, ''Evaluation of fetal distress diagnosis during delivery stages based on linear and nonlinear features of fetal heart rate for neural network community,'' *Int. J. Comput. Appl.*, vol. 156, no. 4, pp. 26– 31, Dec. 2016.

[9] S. Das, K. Roy, and C. K. Saha, ''Establishment of automated technique of FHR baseline and variability detection using CTG: Statistical comparison with expert's analysis,'' *Int. J. Inf. Eng. Electron. Bus.*, vol. 11, no. 1, pp. 27–35, Jan. 2019, doi: 10.5815/ijieeb.2019.01.04.

[10] W.W.Stead,''Clinicalimplicationsandchallengesofartificial intelligence and deep learning,'' *J. Amer. Med. Assoc.*, vol. 320, no. 11, p. 1107, Sep. 2018, doi: 10.1001/jama.2018.11029.

[11] K. H. Miao, J. H. Miao, and G. J. Miao, ''Diagnosing coronary heart disease using ensemble machine learning,'' *Int. J. Adv. Comput. Sci. Appl.*, vol. 7, no. 10, p. 3039, 2016.

[12] M. G. Signorini, N. Pini, A. Malovini, R. Bellazzi, and G. Magenes, ''Integrating machine learning techniques and physiology based heart rate features for antepartum fetal monitoring,'' *Comput. Methods Programs Biomed.*, vol. 185, Mar. 2020, Art. no. 105015.

[13] J. H. Miao, K. H. Miao, and G. J. Miao, ''Breast cancer biopsy predictions based on mammographic diagnosis using support vector machine learning, Multidisciplinary Journals in Science and Technology,'' *J. Sel. Areas Bioinf.*, vol. 5, no. 4, p. 19, 2015.

[14] M. T. Alam, M. A. I. Khan, N. N. Dola, T. Tazin, M. M. Khan,
A. Albraikan, and F. A. Almalki, ''Comparative analysis of differentefficient machine learning methods for fetal health classification,'' *Appl. Bionics Biomechanics*, vol. 2022, pp. 1–12, Apr. 2022.

[15] A. Mehbodniya, A. J. P. Lazar, J. Webber, D. K. Sharma, S. Jayagopalan, K. K, P. Singh, R. Rajan, S. Pandya, and S. Sengan, ''Fetal health classification from cardiotocographic data using machine learning,'' *Expert Syst.*, vol. 39, no. 6, Jul. 2022, Art. no. e12899.

[16] N. M. Fisk and R. P. Smit, ''Fetal growth restriction; small for gestational age,'' in *Turnbull's Obstetrics*, 3rd ed., G. Chamberlain and P. Steer, Eds. Edinburgh, Scotland: Churchill Livingstone, 2001, pp. 197–209.

[17] E.J.QuilliganandR.H.Paul,''Fetalmonitoring:Isitworthit?'' *Obstetrics Gynecol.*, vol. 45, no. 1, pp. 96–100, 1975.

[18] F. G. Esposito, ''Fetal heart rate monitoring and neonatal outcome in a population of earlyand lateonset intrauterine growth restriction,'' *J. Obstetrics Gynaecology Res.*, vol. 45, no. 7, pp. 1343–1351, 2019.

[19] A. Ugwumadu, ''Are we (MIS) guided by current guidelines on intrapartum fetal heart rate monitoring? Case for a more physiological approach to interpretation,'' *BJOG, Int. J. Obstetrics Gynaecol.*, vol. 121, no. 9, pp. 1063–1070, 2014.

[20] C. P. Anaruma, M. Ferreira, C. H. G. Sponton, M. A. Delbin, and
A. Zanesco, ''Heart rate variability and plasma biomarkers in patients withtype 1 diabetes mellitus: Effect of a bout of aerobic exercise,'' *Diabetes Res. Clin. Pract.*, vol. 111, pp. 19–27, Jan. 2016.

[21] F. Andreotti, J. Behar, S. Zaunseder, J. Oster, and G. D. Clifford, ''An opensource framework for stress-testing non-invasive foetal ECG extraction algorithms,'' *Physiological Meas.*, vol. 37, no. 5, pp. 627–648, May 2016.

[22] G. D. Clifford, I. Silva, J. Behar, and G. B. Moody, ''Non- invasive fetal ECG analysis,'' *Physiological Meas.*, vol. 35, no. 8, pp. 1521–1536, Aug. 2014.

[23] T. P. Richardson, M. C. Peters, A. B. Ennett, and D. J. Mooney, ''Polymeric system for dual growth factor delivery,''*Nature Biotechnol.*, vol. 19, no. 11, pp. 1029–1034, Nov. 2001.

[24] D. Maulik, *Doppler Ultrasound in Obstetrics and Gynecology*, I. Zalud, Ed. Berlin, Germany: Springer, 2005, pp. 363–374.

[25] D. Goswami, S. Bhasin, and S. Batra, ''Obstetric & gynecological emergencies,'' in *Wife Goes*, 2011.

[26] P. F. F. Ortiz, ''Automatic fetal distress assessment during labor based on modal and parametricalanalysis of the cardiotocographi crecording,''Ottovon-Guericke-Universität, Magdeburg, Germany, Tech. Rep., 2020

[27] N. Krupa, M. Ma, E. Zahedi, S. Ahmed, and F. M. Hassan, ''Antepartum fetal heart rate feature extraction and classification using empirical mode decomposition and support vector machine,'' *Biomed. Eng. OnLine*, vol. 10, no. 1, p. 6, 2011.M. Ferreira and A. Zanesco, ''Heart rate variability an important approach for the assessment of autonomic modulation,'' *Motriz: Revista de Eduçãsica*, vol. 22, no. 2, p. 38, 2016

[28] M. Ferreira and A. Zanesco, ''Heart rate variability an important approach for the assessment of autonomic modulation,'' *Motriz: Revista de Eduçãsica*, vol. 22, no. 2, p. 38, 2016

[29] V. Paladugu, S. Sreedhar, R. Chitra, S. T. Mannava, S. Sreekumar and J. Mangalakanthi, "Association of CTG diagnosis of intrapartum fetal distress and immediate postpartum acidemia in foetal umbilical artery", J. Obstetrics Gynecol. India, vol. 73, no. 1, pp. 28-35, Feb. 2023.

[30] N. Krupa, M. Ma, E. Zahedi, S. Ahmed and F. M. Hassan, "Antepartum fetal heart rate feature extraction and classification using empirical mode decomposition and support vector machine", Biomed. Eng. OnLine, vol. 10, no. 1, pp. 6, 2011.

[31] M. Ferreira and A. Zanesco, "Heart rate variability an important approach for the assessment of autonomic modulation", Motriz: Revista de Eduçãsica, vol. 22, no. 2, pp. 38, 2016.

[32] M. Schnauer, A. Thomas, S. Morbach, J. Niebauer, U. Schnauer and H. Thiele, "Cardiac autonomic diabetic neuropathy", Diabetes Vascular Disease Res., vol. 5, no. 4, 2008.

[33] D. Hoyer, "Monitoring fetal maturation—Objectives techniques and indices of autonomic function", Physiological Meas., vol. 38, no. 5, pp. R61-R88, May 2017.

[34] E. W. Abdulhay, R. J. Oweis, A. M. Alhaddad, F. N. Sublaban, M. A. Radwan and H. M. Almasaeed, "Monitoring techniques", Biomed. Sci., vol. 2, no. 3, pp. 5367, 2014.

[35] L. Hruban, J. Spilka, V. Chudáček, P. Jankå, M. Huptych, M. Burša, et al., "Agreement on intrapartum cardiotocogram recordings between expert obstetricians", J. Eval. Clin. Pract., vol. 21, no. 4, pp. 694-702, Aug. 2015.