**Predicting Fetal Health from Cardiotocogram Data - A Machine Learning Approach Towards Reducing Child and Maternal Mortality**

Lakshmi Aparna Valiveti, Suneetha Naidu Mekala

[lavaliv@iu.edu](mailto:lavaliv@iu.edu) , [sumekala@iu.edu](mailto:sumekala@iu.edu)

Indiana University Purdue University, Indianapolis, IN46202, USA

1. **Abstract:**

Fetal health monitoring is crucial in obstetrics to prevent perinatal mortality and morbidity. Cardiotocography (CTG), which records fetal heart rates and uterine contractions, is a standard diagnostic technique used during pregnancy and labor to assess fetal well-being and identify distress. Despite its widespread use, CTG interpretation is often hampered by its subjective nature, leading to variability in diagnostic outcomes. This project proposes the use of machine learning (ML) techniques to improve the analysis and interpretation of CTG data. By developing a predictive model that classifies CTG records into three categories—Normal, Suspect, and Pathological—this approach aims to enhance early detection and intervention for fetal distress. This can significantly support healthcare providers in making accurate and timely decisions, especially in under-resourced settings with limited access to specialized diagnostic facilities. Ultimately, the application of ML in fetal health monitoring promises to advance global health objectives by reducing child and maternal mortality and improving prenatal care outcomes. Through this study, we seek to mitigate the challenges associated with traditional CTG analysis and provide a reliable tool for healthcare professionals, thereby improving health outcomes for mothers and infants worldwide.  
**Keywords: f**etal health monitoring, cardiotocography (CTG), fetal distress detection, predictive modeling, perinatal outcomes, diagnostic techniques in pregnancy, histogram, accelerations, decelerations, uterine contractions, fetal movements.

1. **Project Scope:**

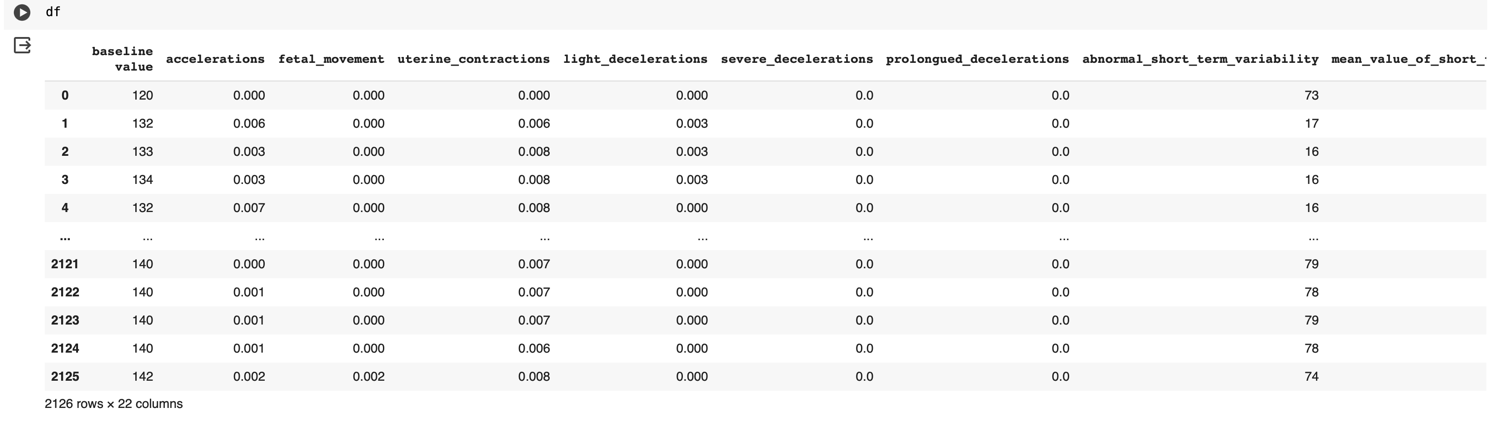
* **Introduction:**  
  Fetal health monitoring is a critical aspect of obstetrics aimed at preventing perinatal mortality and morbidity. Cardiotocography (CTG) is a diagnostic method used in obstetrics that simultaneously records the fetal heartbeat (heart rate) and the uterine contractions during pregnancy, particularly during labor. It is primarily used to monitor fetal well-being and to detect signs of fetal distress. The technique provides vital information that can help healthcare providers decide whether a fetus is healthy or if an immediate delivery is necessary to prevent fetal distress and potential complications. However, the interpretation of CTG traces remains challenging due to its subjective nature, which can lead to inconsistent diagnostic performance. The emergence of machine learning (ML) offers a promising avenue for enhancing the analysis of CTG data. By applying advanced ML techniques, this project aims to develop a predictive model that accurately classifies CTG records into three categories like Normal, Suspect, and Pathological thereby facilitating early detection and timely intervention for fetal distress. This approach not only supports healthcare professionals in making informed decisions but also aligns with global health objectives to reduce child and maternal mortality, particularly in under-resourced settings where access to expert diagnostic facilities is limited. Through the strategic application of machine learning, this study seeks to bridge the gap in prenatal care, ensuring better health outcomes for both mothers and infants.
* **Aim:** The study aims to develop a predictive model using machine learning methods to accurately evaluate fetal health from Cardiotocogram (CTG) data. By categorizing CTG records into three distinct groups—Normal, Suspect, and Pathological—this study aims to furnish healthcare professionals with a dependable tool. Utilizing machine learning techniques, the goal is to facilitate the early identification of fetal health issues, ultimately aiding in decreasing maternal and child mortality rates, especially in areas with limited resources.
* **Research Idea:** This research seeks to identify and analyze the most impactful variables within Cardiotocogram (CTG) data that influence fetal health. Leveraging advanced machine learning techniques, the study aims to pinpoint which specific factors are most predictive of fetal outcomes. By thoroughly understanding these key variables, the project intends to develop a robust predictive model that categorizes fetal health into three classifications—Normal, Suspect, and Pathological. The goal is to equip healthcare professionals with a reliable tool that enhances early detection of potential fetal complications, thereby contributing to the reduction of maternal and child mortality, particularly in under-resourced settings.

1. **Methodology:**

* **Steps:** Data collection, Data extraction, Data cleaning, Data analysis, Data visualization, Statistical Analysis, Machine Learning Models.
* **Programming Language:** Python
* **Tools:** Google Collab, Jupyter Notebook

1. **Data Collection:**

* **Data description:** The dataset has been taken from a website named “Kaggle”, it can be accessed by this link <https://www.kaggle.com/datasets/andrewmvd/fetal-health-classification/data> The dataset contains 2126 rows, 22 columns.



* **Variables:** baseline value, accelerations, fetal\_movement, uterine\_contractions, light\_decelerations, severe\_decelerations, prolonged\_decelerations, abnormal\_short\_term\_variability, mean\_value\_of\_short\_term\_variability, percentage\_of\_time\_with\_abnormal\_long\_term \_variability, mean\_value\_of long\_term\_variability, histogram\_width, histogram\_min, histogram\_max, histogram\_number\_of\_peaks, histogram\_number\_of\_zeroes, histogram\_mode, histogram\_mean, histogram\_median, histogram\_variance, histogram\_tendency, fetal\_health

**5. Data Description:**

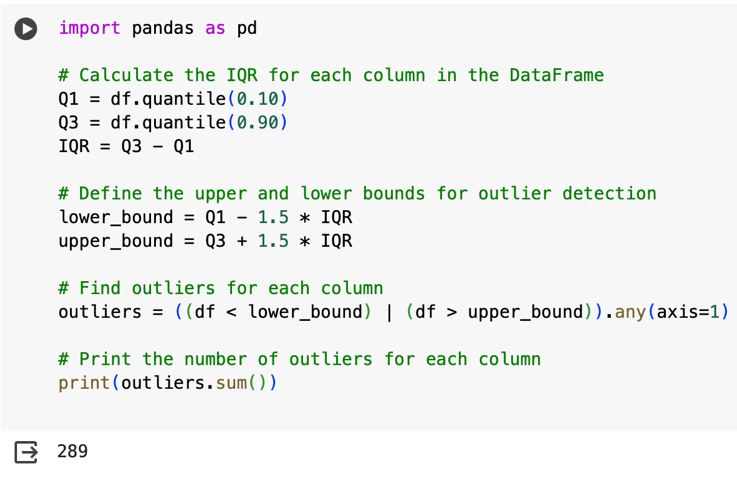
|  |  |  |
| --- | --- | --- |
| **S.No.** | **Variables** | **Description** |
| 1. | baseline value | Baseline fetal heart rate (FHR) |
| 2. | accelerations | Number of accelerations per second |
| 3. | fetal\_movement | Number of fetal movements per second |
| 4. | uterine\_contractions | Number of uterine contactions per second |
| 5. | light\_decelerations | Number of LDs per second |
| 6. | severe\_decelerations | Number of SDs per second |
| 7. | prolongued\_decelerations | Number of PDs per second |
| 8. | abnormal\_short\_term\_variability | Percentage of time with abnormal short term variability |
| 9. | mean\_value\_of\_short\_term\_variability | Mean value of short term variability |
| 10. | percentage\_of\_time\_with\_abnormal\_long\_term\_variability | Percentage of time with abnormal long term variability |
| 11. | mean\_value\_of\_long\_term\_variability | Mean value of long term variability |
| 12. | histogram\_width | Width of the histogram made using all values from a record |
| 13. | histogram\_min | Histogram minimum value |
| 14. | histogram\_max | Histogram maximum value |
| 15. | histogram\_number\_of\_peaks | Number of peaks in the exam histogram |
| 16. | histogram\_number\_of\_zeroes | Number of zeroes in the exam histogram |
| 17. | histogram\_mode | Hist mode |
| 18. | histogram\_mean | Hist mean |
| 19. | histogram\_median | Hist Median |
| 20. | histogram\_variance | Hist variance |
| 21. | histogram\_tendency | Histogram trend |
| 22. | fetal\_health | Fetal health:  1 - Normal  2 - Suspect  3 - Pathological |

**6. Data Cleaning:**

* **Checking for Null Values:** We checked for null values in our dataset, and we don't have any null values.



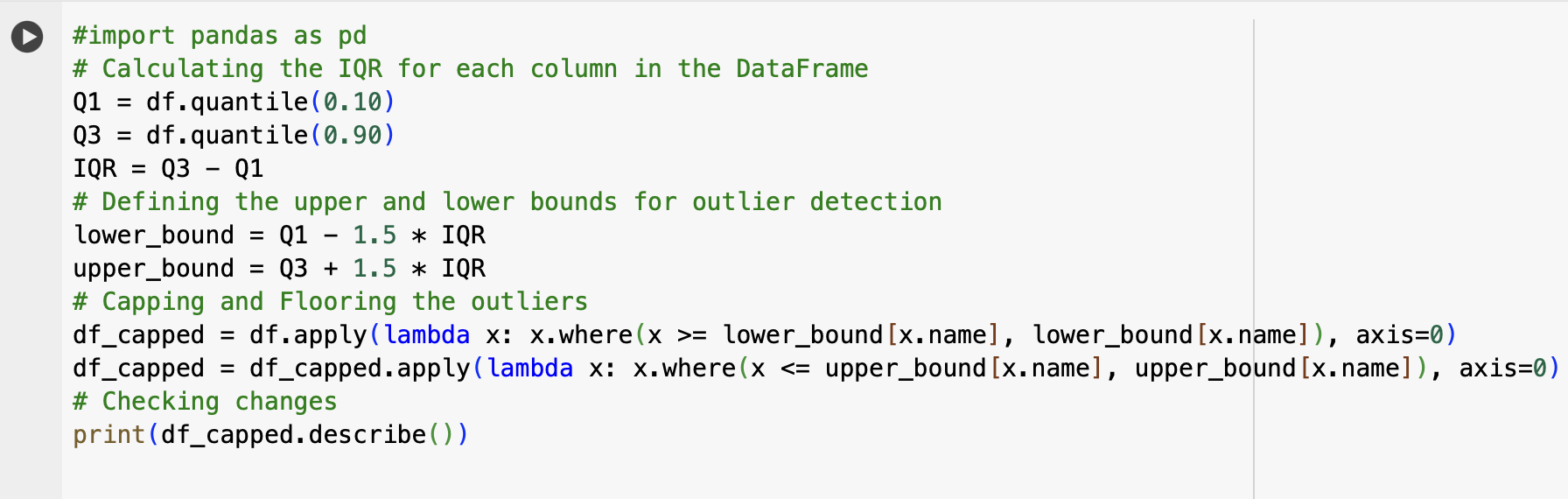
* **Checking for Outliers:** We checked for outliers and represented by using Box-plot method.



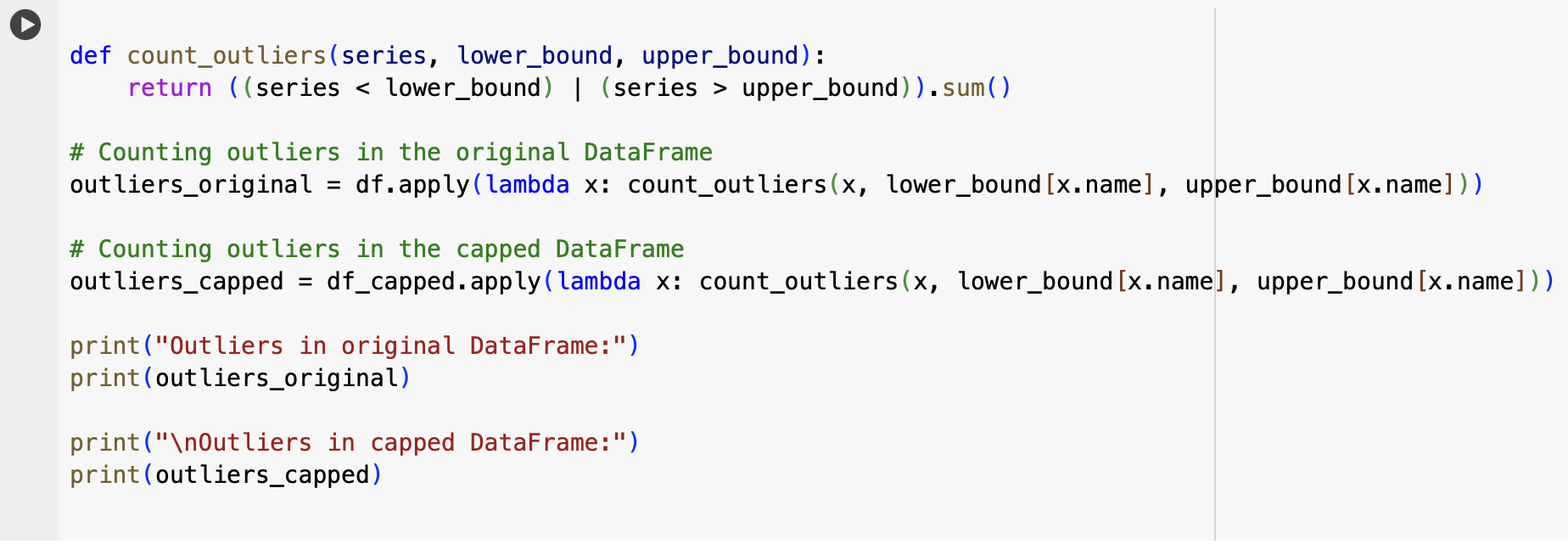
* **Box plot representation:**

|  |  |  |
| --- | --- | --- |
| A diagram of a box plot  Description automatically generated | A diagram of a box with a line and a line  Description automatically generated | A diagram of a boxplot  Description automatically generated |
| A diagram of a box with a line in the middle  Description automatically generated | A diagram of a light  Description automatically generated | A white rectangular object with black text  Description automatically generated |
| A white background with black dots and black text  Description automatically generated | A diagram of a box with a line in the middle  Description automatically generated | A diagram of a graph  Description automatically generated |

* **Treating Outliers by using Capping and Flooring method:** The capping and flooring method is a technique used to handle outliers in the dataset. Capping involves setting a maximum or minimum threshold beyond which data points are considered outliers and are replaced with the nearest value within the threshold. Flooring is the process of replacing outliers with the threshold value itself.



* **Outliers’ representation before and after using Capping and Flooring method:**

  
  
  
A screenshot of a computer code

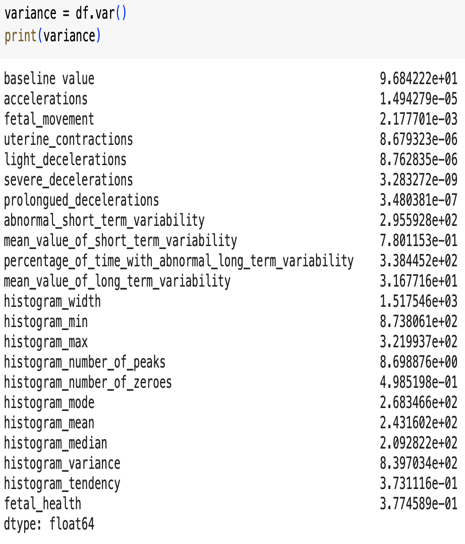
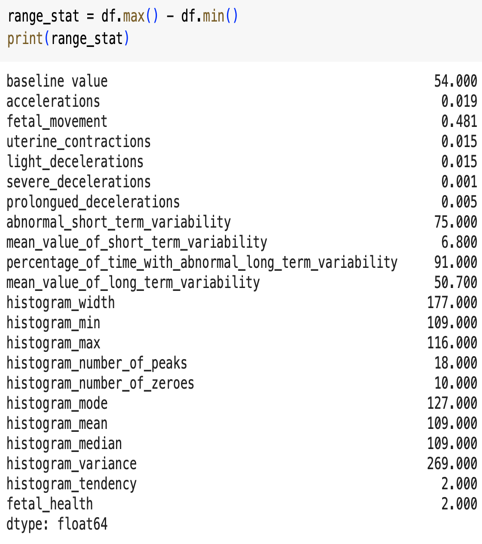
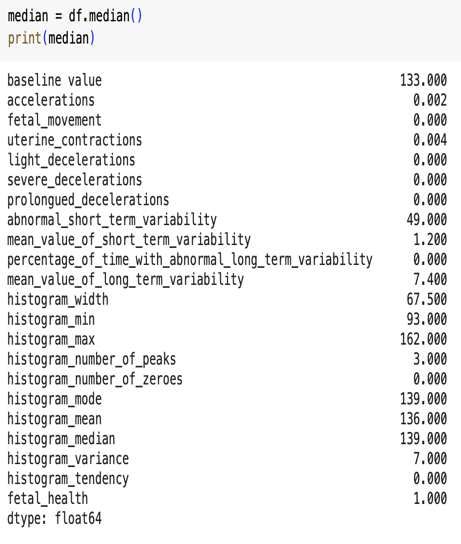
Description automatically generatedA computer screen shot of a code

Description automatically generated

**7. Data Analysis:**

* **Descriptive Analysis:** We used the code **df.describe()** to generate summary statistics for the DataFrame df, transposing the result for better readability. It provides key metrics such as count, mean, standard deviation, minimum, quartiles, and maximum values for each numerical column in the DataFrame. This concise operation simplifies data analysis and reporting by presenting the statistics in a structured and easily interpretable formatA screenshot of a computer screen

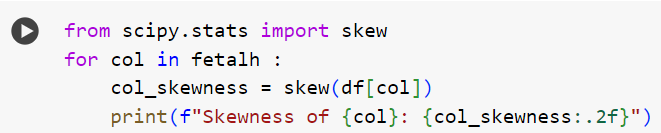
  Description automatically generated

  
**Fig. 6. Median, Range, Variance**

* **Test for Normality:** To know the distribution of the variables in the data, Shapiro-wilk test. The Shapiro-Wilk test assesses if data comes from a normally distributed population. The results show extremely low p-values for all variables, indicating they are significantly different from normal distribution. Therefore, none of the variables appear to be normally distributed based on this test.



* **Checked for Skewness:**



Skewness measures the asymmetry of a distribution. Positive skewness > 0 indicates a longer or fatter tail on the right side, while negative skewness < 0 indicates a longer or fatter tail on the left side. Values closer to zero suggest a more symmetric distribution.

|  |  |
| --- | --- |
| **Positive Skewness:** | **Negative Skewness:** |
| fetal\_movement, severe\_decelerations,  prolongued\_decelerations  percentage\_of\_time\_with\_abnormal\_long\_term\_variability, histogram\_number\_of\_zeroes, histogram\_variance, accelerations, mean\_value\_of\_short\_term\_variability, fetal\_health | histogram\_mode, histogram\_mean, histogram\_median histogram\_tendency |

**8. Statistical tests:**  
  
As we found that the variables in the dataset were not normally distributed, we opted for non-parametric statistical tests. As the comparison is between two or more independent variables with the outcome variable, we have opted for Kruskal-Wallis test.

* **Kruskal-Walli’s test:**  It is a non-parametric test used to determine whether there are statistically significant differences between two or more independent groups. A low p-value < 0.05 suggests that there are significant differences between the groups.

In our results, all p-values are reported as 0.000 or very close to it, except for 'histogram\_max', 'histogram\_number\_of\_zeroes', and 'histogram\_tendency'.

This suggests that for most variables, there are significant differences between the groups being compared. For 'histogram\_max', 'histogram\_number\_of\_zeroes', and 'histogram\_tendency', while the p-values are not as low, they are still below the typical significance level of 0.05, indicating significant differences between groups.



**9. Data Visualization:**

Data visualization involves illustrating data in a graphical style to uncover patterns, trends, and correlations that could influence study conclusions. We presented the data with histograms, pie charts, and bar charts.

We’ve imported Seaborn for high-level statistical visualization, pandas for efficient data manipulation, matplotlib for versatile plotting capabilities, NumPy for numerical computations, and SciPy’s stats module for advanced statistical analysis. Together, these packages provide a comprehensive toolkit for data visualization and statistical exploration in Python.

A screen shot of a computer code

Description automatically generated

**Histogram representation:**

**A group of pink and black bars

Description automatically generated with medium confidence**

**A screen shot of a computer code

Description automatically generated**

**Histogram representation:**

**A graph of different sizes and numbers

Description automatically generated with medium confidence**

**A computer code with text

Description automatically generated**

**Histogram representation:**

**A collage of different types of graphs

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**Histogram representation:**

**A group of gray bars

Description automatically generated with medium confidence**

**A computer code with text

Description automatically generated**

**Bar graph representation of dependent variable – Fetal Health:**

**A graph with numbers and numbers

Description automatically generated**

**A computer screen shot of a code

Description automatically generated**

**Pie chart representation of Fetal health:**

**A blue pie chart with red and white text

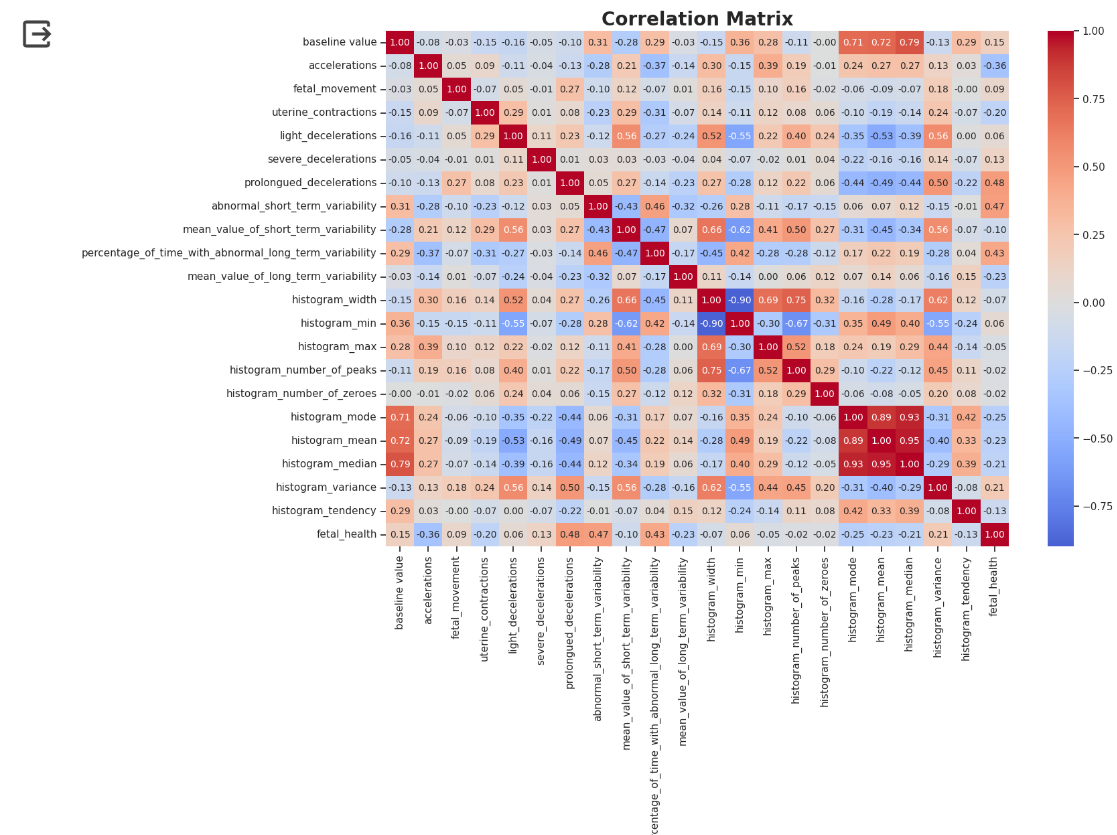
Description automatically generated**

* **Correlation Analysis:**

Before building the prediction model to know the variables which would positively relate to the variable or interest and which would negatively relate to the outcome variable, a correlation test is performed.

A computer screen shot of a computer code

Description automatically generated

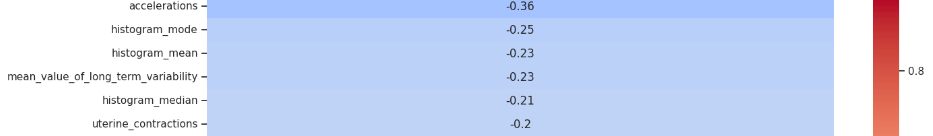


A computer screen shot of a computer code

Description automatically generated

These were the variables which established a more positive correlation with the variable of interest/ prediction variable.

These are the variables which have shown more negative correlation with the variable of interest/prediction variable.



**9. Machine Learning Models:**

As our dataset has the variable of interest in a very small number to increase the sample number and scalability, we performed standard scalar and smote analysis.

* **Standard Scaler Method:** This method transforms the data such that it has a mean of 0 and a standard deviation of 1. This can be achieved by subtracting the mean and then dividing by the standard deviation for each feature. This ensures that all features are on the same scale, which can be important for certain machine learning algorithms that are sensitive to the scale of the input features.

A computer code with black and red text

Description automatically generated

* **Smote Analysis:**

SMOTE (Synthetic Minority Over-sampling Technique) is a method used to address class imbalance in datasets by generating synthetic samples for the minority class. This technique helps to balance class distribution and mitigate the issues caused by imbalanced datasets, such as biased model performance towards the majority class. By going through the results, SMOTE successfully balanced the class distribution of the target variable 'fetal\_health' by generating synthetic samples for the minority classes. Each class now has an equal number of samples, enhancing the robustness and generalization ability of machine learning models trained on the balanced dataset.

A screenshot of a computer program

Description automatically generated

* **Feature Importance:**

Feature importance is a metric that helps us understand the relative importance or contribution of each feature in the dataset towards making predictions. It provides insights into which features most strongly influence the output of the model.

A screenshot of a computer program

Description automatically generated

**Results:**

From the graph, we can see that 'percentage\_of\_time\_with\_abnormal\_long\_term\_variability', "abnormal\_short\_term\_variability", 'histogram\_mean', 'prolongued\_decelerations', and , 'mean\_value\_of\_short\_term\_variability', ‘histogram\_median’, ‘accelerations’are the most significant features, implying that they have the greatest impact on the model's predictions.

Among this, these are the features we picked for machine learning model building, 'percentage\_of\_time\_with\_abnormal\_long\_term\_variability', 'abnormal\_short\_term\_variability', 'histogram\_mean', 'prolongued\_decelerations', 'mean\_value\_of\_short\_term\_variability', 'severe\_decelerations'

A graph of blue and white text

Description automatically generated

* **Random Forest Classifier:**

Random Forest is an ensemble learning method that builds multiple decision trees during training and merges their predictions to improve accuracy and robustness. It is trained using a subset of selected features identified as important. The classifier is then evaluated on a test set using metrics like accuracy, precision, recall, and F1-score.

**Results:** The model achieved an accuracy of 91.08% on the test set, with high precision and recall values for each class. This suggests that the model is performing well in accurately classifying fetal health based on the selected features.

A screenshot of a computer program

Description automatically generated

A blue and white squares with black numbers

Description automatically generated

A screenshot of a computer

Description automatically generatedA graph of a multi-class curve

Description automatically generated

* **KNN Classifier:**

K-Nearest Neighbors (KNN) is a simple and effective classification algorithm that classifies data points based on the majority class of their nearest neighbors. The model is trained using a subset of selected features identified as important. The classifier is then evaluated on a test set using metrics like accuracy, precision, recall, and F1-score. The KNN classifier achieved an accuracy of 87.09% on the test set, with varying precision, recall, and F1-score values for each class. The confusion matrix visualizes the performance of the classifier, showing the number of correctly and incorrectly classified instances for each class.

A screenshot of a computer program

Description automatically generatedA graph with numbers and a blue square

Description automatically generatedA graph of a line graph

Description automatically generated with medium confidence

* **Gradient Boost Classifier:** Gradient Boosting is an ensemble learning technique that builds a strong predictive model by combining multiple weak learners, typically decision trees, in a sequential manner.

The model is trained using a subset of selected features identified as important. The classifier is then evaluated on a test set using metrics like accuracy, precision, recall, and F1-score. In this case, the Gradient Boosting Classifier achieved an accuracy of 89.20% on the test set, with high precision and recall values for each class. The confusion matrix visualizes the performance of the classifier, showing the number of correctly and incorrectly classified instances for each class.

A screenshot of a computer program

Description automatically generated

A graph with numbers and a bar

Description automatically generatedA graph of a curve

Description automatically generated with medium confidence

* **Decision Tree Classifier:**

Decision Tree is a versatile and interpretable machine learning algorithm used for classification and regression tasks. It builds a tree-like structure where each internal node represents a feature, each branch represents a decision based on that feature, and each leaf node represents the outcome or prediction.

The model is trained using a subset of selected features identified as important. The classifier is then evaluated on a test set using metrics like accuracy, precision, recall, and F1-score. In this case, the Decision Tree Classifier achieved an accuracy of 89.91% on the test set, with varying precision, recall, and F1-score values for each class. The confusion matrix visualizes the performance of the classifier, showing the number of correctly and incorrectly classified instances for each class.

A screenshot of a computer code

Description automatically generatedA graph with numbers and a bar

Description automatically generatedA graph of a line

Description automatically generated with medium confidence

* **Comparison of Machine Learning Models:**

The comparison four machine learning models: Random Forest, KNN, Gradient Boosting, and Decision Tree, using selected features from the dataset. The models were trained on a subset of selected features from a dataset and their accuracies were evaluated.

A screenshot of a computer program

Description automatically generated

A graph showing different colors of different sizes

Description automatically generated

**10. Results and Conclusion:**

1. From the analysis, the Random Forest classifier demonstrates the highest accuracy at 91.08%, making it the most effective model among those tested for this specific dataset and feature set.
2. The KNN model shows the lowest accuracy at 87.09%. Gradient Boosting and Decision Tree models deliver similar performance with accuracies just below 90%.
3. It is the ensemble approach of Random Forest that proved to be most effective in handling the complexity of the dataset.
4. The application of such accurate models can aid in early detection of potential health issues, ultimately contributing to the improvement of maternal and fetal

outcomes.

**References:**

<https://www.kaggle.com/datasets/andrewmvd/fetal-health-classification/data>

Hasan, B., Hoodbhoy, Z., Noman, M., Shafique, A., Nasim, A., & Chowdhury, D. (2019). Use of machine learning algorithms for prediction of fetal risk using cardiotocographic data. *International Journal of Applied and Basic Medical Research*, *9*(4), 226. https://doi.org/10.4103/ijabmr.ijabmr\_370\_18