Fetal Distress Prediction Based on Cardiotocographic (CTG) Data

Suyashi Singhal, Harshita Gupta, Ayush Mahant, Rasagya Shokeen
Deparment of Computer Science
Indraprastha Institute of Information Technology, Delhi
{suyashi19478, harshita19467, ayush19353, rasagya19088}@iiitd.ac.in

Abstract

Cardiotocography(CTG) is a monitoring technique used to determine a fetus's healthy being by simultaneously recording the fetal heart rate and the mother's uterine contractions. The recorded CTG data can provide obstetricians with vital information that can determine the well-being of the fetus and the mother. However, visual inspection of such data might not be very reliable, and hence we require additional ways of assessing and evaluating fetal well-being. Advanced machine learning algorithms should help us analyse the CTG data and predict the fetal state. Our main aim is to develop a machine learning model that can identify high-risk fetuses accurately comparable to highly trained medical professionals. We hope that this would play a significant role in reducing fetal mortality and congenital disabilities globally. We have used a CTG dataset containing 21 features and 2112 data points obtained from the UCI Machine Learning Repository. We employ various robust machine learning models to classify the fetal state into three classes: Normal, Suspect, and Pathogenic. Trained on 3-fold cross-validation, we have analysed the classifiers' successes using a variety of performance metrics calculated from the confusion matrix. According to our analysis, almost all the machine learning models have shown satisfactory performance; however, random forest and decision trees have proven to be the most efficient in classification with an accuracy of 97.30% and 95.53%, respectively. Furthermore, to improve the accuracy of the models, we have used techniques like random oversampling, principal component analysis, and extensive data visualisation. The code and trained models are available at https://github.com/suyashi912/MLproject-Cardiotocography

Keywords **- Cardiotocography, Machine learning models, Classification, Random oversampling**

1. Introduction

The number of fetal (unborn baby in the mother's womb) and maternal deaths every year worldwide is staggering. Undetected fetal abnormalities can progressively worsen, leading to permanent damage to the fetus and even death. However, early intervention can potentially be life-saving for both the mother as well as the child.

Cardiotocography is one such technique that can concurrently monitor the fetal heart rate (FHR) and uterine contractions(UC). It can detect abnormal fetal state and movements, which could be early symptoms of dangerous fetal distress conditions like intrapartum hypoxia/asphyxia. If undiagnosed, this oxygen deficiency can alter fetal physiology, resulting in probable brain damage and even fetal death. A cardiogram is a machine used to perform cardiotocography. It is an electronic fetal monitor that employs the Doppler ultrasound effect using two transducers placed externally on the pregnant woman's abdominal wall to record the FHR and UC signals. A time-scaled running line graph depicts the intrauterine pressure measured from the abdominal wall tensions. Trained clinicians and obstetricians can interpret the CTG data. However, this method has the drawback that visual inspection of the data is often unreliable and can vary from person to person leading to inconsistent interpretations. Over 50% of fetal deaths are due to this inconsistency in pattern recognition and failure in receiving a timely intervention.

Therefore, integrating computerised machine learning methods with obstetrician interpretations can prove indispensable in predicting fetal distress conditions and providing early intervention. In this study, our objective is to employ such machine learning classifiers that can provide good performance over unseen data and improve the cardiotocography technique. We use a dataset obtained from the UCI Machine Learning Repository. After performing appropriate feature selection and analysis, we select 21 core features that influence the fetal state. We thus have 2112 data samples out of the 2126 data points after data preprocessing. We compare and contrast various machine learning classifiers using performance and evaluation metrics like accuracy, precision, recall and f1 score. Furthermore, we use GridSearchCV to pick the best hyperparameters for each model. It is worth mentioning that the imbalanced distribution of data can prove to be a problem in classification. Thus, we utilize a novel approach of random oversampling to increase the data samples of minority classes and employ principal component analysis for feature selection.

This report summarises the related works of literature we have read, the machine learning models trained by us, and the consequent analysis and results obtained from the model. We also draw conclusions based on our study.

2. Literature Survey

Hoodbhoy et al. [1] study the precision of the machine learning algorithm technique on the CTG dataset, aiming to identify high-risk fetuses as accurately as highly trained medical professions. For data balancing SMOTE technique is used that avoids overfitting on skewed classes. Out of the ten machine

learning models, the XGBoost method had the highest overall accuracy of about 93%. However, it did have the drawback of having a comparatively low sensitivity in the "Suspect" state compared to the "Pathologic" condition.

Zhang et al. [3] present a novel approach for distinguishing fetal states into normal and pathological categories instead of classifying the fetal states into three classes. This paper uses the CTG dataset from UCI with 1831 groups, including 21 features and one label. Principle Component Analysis (PCA) performs dimensionality reduction and feature selection, improving accuracy and computational time. It struggles with the problem of outliers as the dataset used has no outlier values, which is not justifiable in the real world. [3]

In this paper[2], the authors present both R and Python machine learning techniques for performance analysis. The study shows that classification studies should be accountable to models and parameter settings and the tools used. Four different types of feature selection based on feature correlations and various models are employed for this study.

Unlike other studies, the authors in paper [4] also evaluated extreme learning machines[ELM] algorithm with five different activation functions apart from Random Forest Classifier, Support Vector Machines, Artificial Neural Network, and Radial Based function network.

Subha et al. select features using techniques like Gain Ration Attribute Evaluation, Relief Attribute Evaluation, and Symmetrical Uncertainty Attribute Evaluation to obtain a dataset with reduced features and thus increase model accuracy.[5]

3. Data Preprocessing and Visualisation

3.1. Dataset Description

We have used the Carditocography raw data from the UCI Machine Learning Repository (SisPorto). The data consists of 2126 data samples and 28 features. It gives two types of classifications : with respect to a morphologic pattern(10 classes) and to a fetal state(3 classes). We classify according to the fetal state : N, S, P and use 21 features after performing feature selection.

3.2. Data Visualisation

We plotted the correlation heat map (Figure 1) of the features and selected the highly correlated ones. We also plotted a boxplot (Figure 2) of the features to visualize distribution of a variable using a five number summary which contains the minimum value, first quartile(25%), median(50%), third quartile(75%) and maximum value.

3.3. Preprocessing

3.3.1 Feature selection and cleaning

We have dropped 7 features namely: Filename, Date, Segmentation File, b (Start time), e (End time), LBE(baseline value - medical expert) and DR(repetitive decelerations) since these features did not seem relevant to the fetal state. Furthermore we dropped the rows with null values and removed duplicate data samples. We thus obtained 2112 data samples and 21 features.

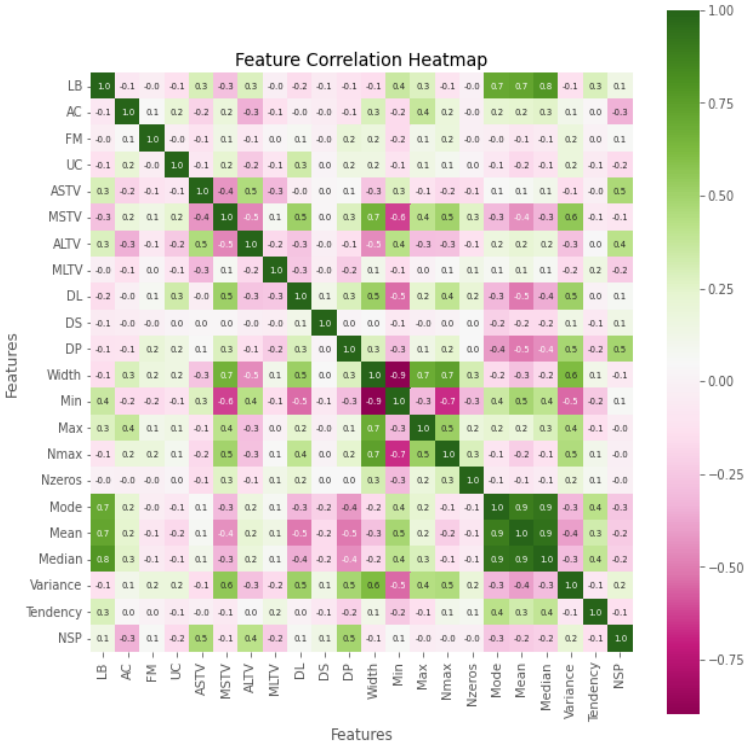

Figure 1. Heatmap of correlation between features in provided dataset

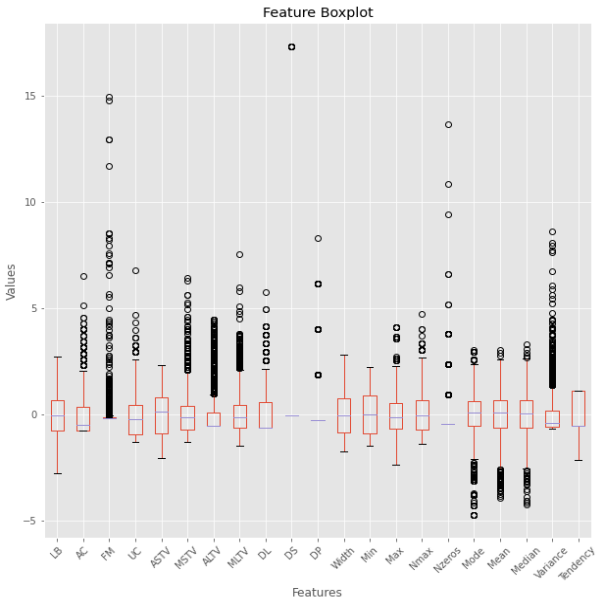

Figure 2. Features box plot for displaying distribution of data based on ("minimum",first quartile, median, third quartile, and "maximum")

3.3.2 Dimensionality Reduction (PCA)

We performed dimensionality reduction using the Principal Component Analysis techniques on various kernels to reduce the dimension of the data. We noticed that our choice of 21 features was appropriate. Interestingly, we observed that each kernel created two distinct groups of "Pathologic" class : one is close to "Suspect" and one is far from it. However, the data is not linearly separable as observed from the plots of various kernels. We have

added the plot for the "Linear" kernel (Figure 3) for reference. All other kernels give similar plots.

3.3.3 Normalization

In order to give equal weights to each feature in the dataset so that no single variable steers model performance in one direction, we performed data normalization. We have used the Min Max Scaling method that squishes the feature values between 0 and 1.

3.3.4 Oversampling data

The division of 2112 samples into 3 classes is as follows: 1646 normal, 292 suspect and 174 pathologic. This indicates that the dataset is imbalanced across the given classes. Thus, we used the random oversampling technique to avoid overfitting of the machine learning model on skewed classes by increasing the data samples of the classes with minority instances. It randomly selects minority class samples and randomly adds them to the dataset to balance it. We got 3588 data samples after oversampling.

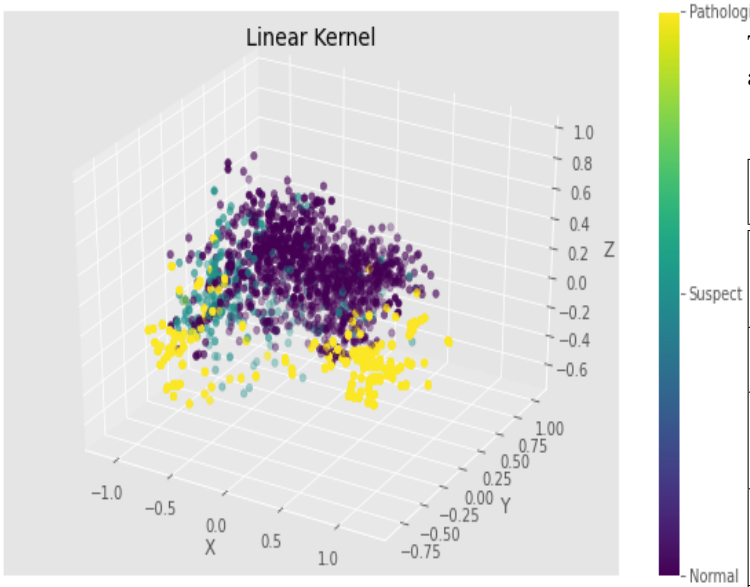

Figure 3. 3d Linear kernel plot for NSP group representation

loss function to cross-entropy loss and predict probability distribution to a multinomial probability distribution.

2. *Naive Bayes*: It uses probability theory to classify data. Naive Bayes classifier algorithms make use of Bayes' theorem. All attributes of a data point under study are considered to be independent of each other.

3. *Decision Trees*: It is a non parameterized supervised learning technique with a pre-defined target variable and is often used in classification problems.

4. *Random forest*: It is an ensemble learning method where many decision trees are constructed at training time. The output of the random forest is the class selected by the majority of the trees.

5. *K - Nearest Neighbors*: It is based on a supervised learning technique that calculates the nearest k neighbours for each data point. The majority label among those data points is returned as the predicted class.

The following table depicts the values of the hyperparameters after tuning them using GridSearchCV -

Classifier Algorithm	**Optimal Parameters**
Logistic Regression	C=1291.54, max_iter=5000, penalty='l2', solver='saga', multi_class = 'multinomial', random_state = 0
Naive Bayes	estimator = GaussianNB(), 'var_smoothing' = 0.18376
KNN	n_neighbors= 2, metric = 'euclidean', weights= 'distance', algorithm = 'ball_tree'
Decision Tree	max_depth = 12, criterion = "entropy", max_features="auto" splitter = "best", random_state=0
Random Forest	n_estimators = 135, criterion = "entropy", max_features = 'auto'

Table 1 : **Tuned Hyperparameters**

4. Methodology

4.1. Model Details

We have split the above dataset into a training and testing set using a 70:30 stratified split. After that, we trained the data on 3 fold cross-validation. Hence we achieved a train:validation: test split of 47:23:30. After dividing the dataset, we chose some supervised learning models to train and test on the dataset. We also performed hyperparameter tuning using GridSearchCV and chose the best model for training and testing. The description of the different models that we employed are as follows:

1. *Logistic Regression*: It is a statistical model that uses a logistic function to model a binary dependent variable in its basic form. We have used the multinomial logistic regression to perform multi-class classification by changing the

4.2. Performance metrics

We used accuracy, precision, recall and F1-score as the evaluation metrics to test our models :

1. *Accuracy*: Accuracy measures the overall efficiency of a classifier. We require that most of the fetal states are classified correctly for a good performance. It is worth noting that even predicting normal state correctly is important because wrong predictions lead to more cesarean sections. [TP / (TN +FP+ FN +TP)]

2. *Precision*: It is the ratio of true positives to the total of the true positives and false positives. Here, it is the measure of the number of fetal states classified correctly out of all the positive samples. [TP / (TP + FP)]

3. *Recall*: It is the ability of a classifier to categorize positively labeled data. Hence, for all fetuses, it tells us how many we correctly classified. [TP / (TP + FN)]

4. *F1 score*: It is the harmonic mean between precision and recall. High F1 score enables us to get a good trade-off between precision and recall. [TP / (TP + 0.5 (FP + FN))]

5. Results and Analysis

The experimental results of the machine learning models on the testing data are depicted in the given table and histogram. The precision, recall and F1 score metrics are the weighted averages of the corresponding scores of the three classes i.e. N-Normal, P-Pathologic and S-Suspect.

Classifier Algorithm	Accuracy	Precision	Recall	F1 score
Logistic Regression	91.72	91.66	91.72	91.68
Naive Bayes	73.14	88.68	73.14	77.45
KNN	95.26	94.96	95.26	95.01
Decision Tree	95.54	95.44	95.53	95.46
Random Forest	97.30	97.21	97.30	97.19

Table 2 : **Evaluation Metrics**

The highest accuracy with a similar performance of the classification algorithms has been observed both in Decision Trees (95.54%) and Random Forest (97.30%) Classifiers followed by KNN (95.26%) and Logistic Regression (91.72%) models, and Gaussian Naive Bayes gives the least accuracy (73.14%).

Logistic Regression gives a decent performance as compared to the other models. We have used the multinomial logistic regression to perform multi-class classification, which offers better performance over the One-vs-All method. Gaussian Naive Bayes has the worst accuracy since it assumes no dependency between attributes which is not the case as the heat map depicts a high correlation between various features.

Decision trees and Random Forests give the best accuracy and precision. The decision tree can easily handle high dimensional non parameterized data and works well with non-linearly separable patterns. Hence, it performs well after pruning and gives an accuracy of 95.54%. It has the best performance with a depth that falls between 12 and 19. Random forest enhances the performance of decision trees further. It is an ensemble method that combines the output of multiple unpruned decision trees and makes a prediction based on the majority vote. It gives an accuracy of over 97.30% and achieves similar precision and recall scores as well.

The K-Nearest Neighbours model works well because it operates on the correlation of features. We have correlated features like Minimum and Maximum values of FHR signals and several other derived variables related to the FHR histogram, which work well with the KNN machine learning model. Furthermore, the KNN model favors a noiseless dataset, and the fact that we have a clean dataset enhances its predictions.

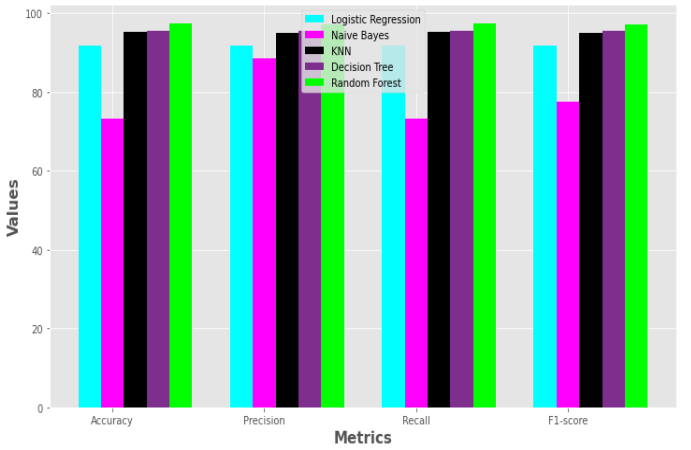

Figure 4. Histogram showing comparison of accuracy, precision, recall and f1 score of the model techniques used in the study

6. Conclusion

Through this study it is shown that preprocessing, extensive machine learning model techniques, and tools used play a vital role in classification scores analysis. The machine learning model techniques used are Multinomial Logistic regression, Gaussian Naive Bayes, Decision trees, Random forests and K-Nearest Neighbors with 3-fold cross validation. Random Forests and Decision Trees give the best accuracies. Our methodology not just includes data cleaning and normalization but also incorporates advanced feature selection and engineering methods like Principal Component Analysis used for Dimensionality Reduction. Finally, we have selected the relevant features and observed that the data is not linearly separable even after performing dimensionality reduction. We have achieved almost perfect accuracy, precision, recall, and F1 score metrics of around 95-98% in 3 out of 5 models, which reiterates that the feature engineering performed by us has performed well. CTG requires expert interpretation unavailable in remote areas, leading to unavailability of dataset from women in such areas, which leads to a problem in identification of proper set of classification of normal, suspect, and pathological cases in India. Hence the dataset does not consider differences in sociodemographic characteristics of pregnant women and some other relevant features like age, nutritional status and so on. This is one of the drawbacks of this study.

Work Left We are right on schedule in terms of the work done and have covered everything that we needed to do till week 5. The work left includes implementing models like Bagging and Boosting, SVM, MLP, their hyperparameter tuning and result anaysis as well as final presentation of the project.

7. Individual Tasks

Tasks	**Team Member/s**
Data preprocessing, DT, KNN, LR	Suyashi
Principal Component Analysis, RF, KNN, LR	Ayush
Plotting maps, RF, NB, LR	Harshita
Plotting maps, DT, NB, LR	Rasagya

References

[1] Z. Hoodbhoy, Md. Noman, A. Shafique, Ali Nasim, D. Chowdhury, B. Hasan, *Use of Machine Learning Algorithms for Prediction of Fetal Risk using Cardiotocographic Data*, 2019

[2] S.C.R Nandipati, C. XinYing, *Classification and Feature Selection Approaches for Cardiotocography by Machine Learning Techniques*, 2020

[3] Y. Zhang, Z. Zhao, *Fetal State Assessment Based on Cardiotocography Parameters Using PCA and AdaBoost*, 2017

[4] Z.Cömerta, A.F. Kocamazb, *Comparison of Machine Learning Techniques for Fetal Heart Rate Classification*, 2016

[5] V.Subha, Dr.D.Murugan, Jency Rani, Dr.K.Rajalakshmi , *Comparative Analysis of Classification Techniques using Cardiotocography Dataset*, 2013