

Fetal wellness Detection based on Machine learning techniques

Chetan¹, Dr. Mohammed Rafi²

¹II Semester M.Tech , DOS in CSE, University BDT College of Engineering [A Constituent College of VTU Belagavi]Davanagere-577004,Karnataka,India

²Professor, DOS in CSE, University BDT College of Engineering [A Constituent College of VTU Belagavi] Davanagere-577004, Karnataka, India

Abstract: *Cardiotocography (CTG) is a non-invasive technical means to record the fetal heart beat and uterine contractions during pregnancy. CTG monitoring is widely used to assess fetal wellbeing. This technique can be used to interpret the fetal health conditions which are very crucial and has high medical impact during the labor period. It helps in making informed decisions by the doctors about the need of surgical interventions. Machine learning on the other hand is a powerful technology which focuses on study of the computer algorithms that provide the ability to learn and improve over experience. The objective is to allow the computers to learn themselves automatically without human intervention. Combining the power of machine learning to develop intelligent system which can automatically detect the status of the fetus will help in reducing the health complications in future. This also reduces the infant mortality at birth and also stillbirths. It will also facilitate the use of telemedicine and tele monitoring for maternal care.*

In this study we used three machine learning algorithms logistic regression, random forest and support vector machine to build prediction models which can predict fetal health leveraging the cardiotocography data. Data preprocessing techniques such as outlier removal, feature selection were done to develop high accuracy models. Hyper parameter tuning was done on each model to optimize their performance. At the end, a comparative analysis between the built models was performed. Comparatively the best model in terms of detecting all the pathologic and suspected state and precisely detecting the normal state was selected with an accuracy of 95.3%. The findings demonstrate improvement on the current literature and possess potential to be incorporated in clinical setup.

Keywords: *Cardiotocography, Machine learning, logistic regression, random forest.*

1. INTRODUCTION

The purpose of CTG recordings is to identify when there is concern about fetal well-being to allow interventions to be carried out before the foetus is harmed. The focus is on identifying fetal heart rate (FHR) patterns associated with inadequate oxygen supply to the foetus. In general, FHR patterns classified as normal are a reliable indication of fetal well-being. Up to 50% of FHR patterns classified as pathological reflect physiological changes and can therefore be classified as false positives (false pathological). This can lead antepartum to increased numbers of induced births and higher numbers of operative deliveries. The most common causes of false positives are when certain disturbances and influencing variables are disregarded (e.g., fetal behavioural states, age of gestation), failure to use additional complementary evaluation methods, uncertainty of interpretation, and inconsistent threshold values and assessment modalities [1].

1.1 Machine learning

Machine learning involves computers discovering how they can perform tasks without being explicitly programmed to do so. It involves computers learning from data provided so that they carry out certain tasks.

Machine learning approaches are traditionally divided into three broad categories, depending on the nature of the "signal" or "feedback" available to the learning system

Supervised learning: The computer is presented with example inputs and their desired outputs, given by a "teacher", and the goal is to learn a general rule that maps inputs to outputs.

Unsupervised learning: No labels are given to the learning algorithm, leaving it on its own to find structure in its input. Unsupervised learning can be a goal in itself (discovering hidden patterns in data) or a means towards an end (feature learning).

Reinforcement learning: A computer program interacts with a dynamic environment in which it must perform a certain goal (such as Autonomous driving or playing a game against an opponent). As it navigates its problem space, the program is provided feedback that's analogous to rewards, which it tries to maximize.

2. LITERATURE REVIEW

According to the guidelines fetal heart rate and fetal circulation determine fetal supply. Physiologically, this system is regulated by the nervous system, which controls the heart rate. Medullary centers controlled by pressoreceptors and chemoreceptors and local metabolic processes influence regulation. The continuous adjustment of arterial and venous blood pressure, cardiac output and vascular resistance is responsible for maintaining adequate metabolic capacity. Deviations from this steady state lead to changes in fetal heart rate (FHR) such as accelerations, variability, tachycardia, decelerations and bradycardia. The complexity of the fetal response to various disturbances and influencing variables often results in misinterpretations of the FHR.

Fetal heart rate is classified into

- baseline fetal heart rate (baseline rate),
- accelerations,
- decelerations,
- oscillations,
- oscillation amplitude (range),
- Long-term oscillations (oscillation rate).

The authors [1] recommend classifying CTG readings into normal/ suspicious/pathological (NSP classification)The below table depicts the same.

Table 1: Evaluation of individual FHR parameters

Parameter	Baseline rate (bpm)	Range (bpm)	Decelerations	Accelerations
Normal	110-160	≥ 5	none ¹	sporadic ²
Suspicious	100-109 161-180	< 5 ≥ 40 minutes > 25	Early/variable dec. individual prolonged dec. up to 3minutes	present, periodical occurrence (with every contraction)
Pathological	< 100 > 180 sinusoidal ³	< 5 > 90minutes	atypical variable dec. late dec. isolated prolonged dec. > 3 minutes	absent > 40minutes (significance still unclear, evaluation questionable)

1 FHR deceleration amplitude ≥ 15 bpm, duration ≥ 15 seconds

2 FHR acceleration amplitude ≥ 15 bpm, duration ≥ 15 seconds

3 sinusoidal FHR: ≥ 10 bpm, duration ≥ 10 minutes

The dataset used for the detection of fetal wellness was sourced from 2126 fetal cardiocograms (CTGs) and these were processed automatically and their diagnostic features were measured. The CTGs were also classified by three expert obstetricians and a consensus classification label assigned to each of them. Classification was both with respect to a morphologic pattern (A, B, C. ...) and to a fetal state (N, S, P) these data were collected at University Hospital of Porto in Portugal by (Ayres-de-Campos, Bernardes et al. 2000).

Cardiocograms are also processed by SisPorto (Ayres-de-Campos et al. 2000) and 21 features are extracted automatically.

In [6], Muhammad Arif have proposed a method to classify using random forest classifier and also studied the impact of reduced set of the features by selecting features by defining an appropriate threshold value below which all features are rejected.

3. PROPOSED SYSTEM

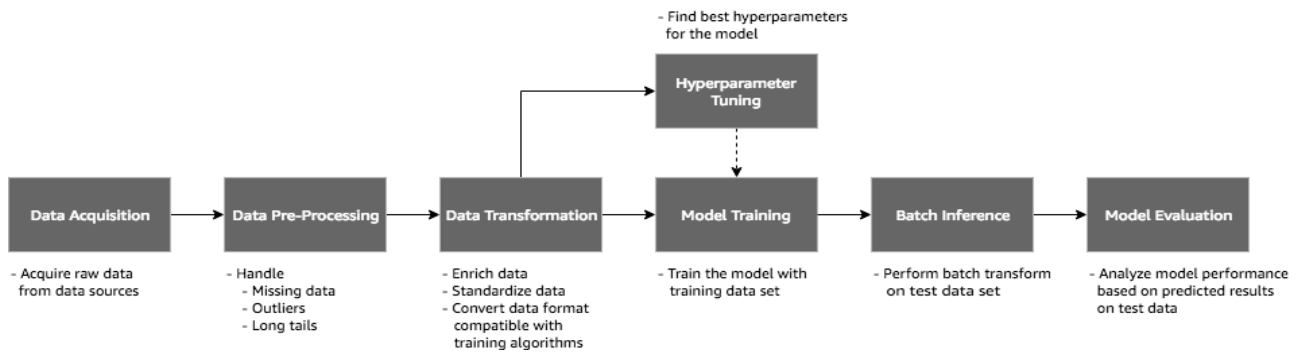


Figure 1: Workflow representing the end to end activities in building a detection system.

This workflow performs the following tasks:

- Data preprocessing: Extract and pre-process data from source to prepare the training data. Techniques such as outlier removal, feature selection was done to develop high accuracy models.
- Prepare training data: To build the system we will split the data into training data and test data and prepare the data for the training.
- Tune the model hyper parameters: It is done on each model to optimize their performance. At the end, a comparative analysis between the built models was performed.
- Model Evaluation: This evaluates a machine learning model. This is dealt in the section in more detail.

3.1 Machine learning algorithms:

Logistic Regression:

Logistic regression is a classification algorithm used to assign observations to a discrete set of classes. Unlike linear regression which outputs continuous number values, logistic regression transforms its output using the logistic sigmoid function to return a probability value which can then be mapped to two or more discrete classes.

Based on the number of categories, Logistic regression can be classified as:

1. Binomial: target variable can have only 2 possible types: “0” or “1” which may represent “win” vs “loss”, “pass” vs “fail”, “dead” vs “alive”, etc.
2. Multinomial: target variable can have 3 or more possible types which are not ordered (i.e. types have no quantitative significance) like “disease A” vs “disease B” vs “disease C”.
3. Ordinal: it deals with target variables with ordered categories. For example, a test score can be categorized as: “very poor”, “poor”, “good”, “and very good”. Here, each category can be given a score like 0, 1, 2, 3.

Logistic regression models the data using the sigmoid function. The below figure depicts the hypothesis of the sigmoid function:

$$g(z) = \frac{1}{1 + e^{-z}}$$

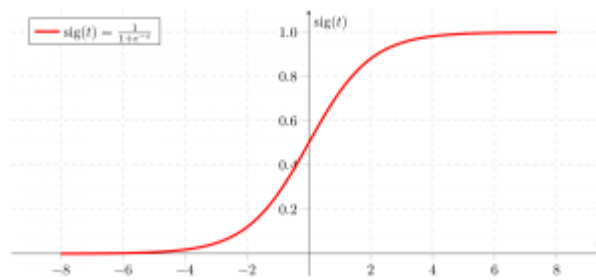


Figure 2: Representing the sigmoid function.

Support Vector Machine (SVM):

A support vector machine (SVM) is a non-probabilistic binary linear classifier. The nonprobabilistic aspect is its key strength. This aspect is in contrast with probabilistic classifiers such as the Naïve Bayes. That is, an SVM separates data across a decision boundary (plane) determined by only a small subset of the data (feature vectors). The data subset that supports the decision boundary are aptly called the support vectors. The remaining feature vectors of the dataset do not have any influence in determining the position of the decision boundary in the feature space. In contrast with SVMs, probabilistic classifiers develop a model that best explains the data by considering all of the data versus just a small subset.

SVM for Classification:

Classification in SVM is an example of Supervised Learning. Known labels help indicate whether the system is performing in a right way or not. This information points to a desired response, validating the accuracy of the system, or be used to help the system learn to act correctly. A step in SVM classification involves identification as which are intimately connected to the known classes. This is called feature selection or feature extraction. Feature selection and SVM classification together have a use even when prediction of unknown samples is not necessary. They can be used to identify key sets which are involved in whatever processes distinguish the classes [3].

Multiclass SVM:

Researchers devised various strategies to address the multi classification problem, including

- One-versus-the-rest: It is probably the earliest SVM multiclass implementation and is one of the most commonly used multiclass SVMs. It constructs c binary SVM classifiers, where c is the number of classes. Each classifier distinguishes one class from all the others, which reduces the case to a two-class problem
- pair-wise classification: This builds $c(c - 1)/2$ binary SVMs, each of which is used to discriminate two of the c classes only and requires evaluation of $(c - 1)$ SVM classifiers
- Multiclassification formulation: It has the most compact form, as it optimizes the problem in a single step. The decision function is the same as that of the one-against-all(OAA) technique. The multiclassification objective function constructs c two-class rules, and c decision functions solve the following constraints

Random Forest:

Random forest is a supervised learning algorithm. The “forest” it builds, is an ensemble of decision trees, usually trained with the “bagging” method. The general idea of the bagging method is that a combination of learning models increases the overall result. This algorithm builds multiple decision trees and merges them together to get a more accurate and stable prediction.

A random forest is a meta estimator that fits a number of decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting[4]

One big advantage of random forest is that it can be used for both classification and regression problems, Random forest have nearly the same hyper parameters as a decision tree or a bagging classifier. Fortunately, there's no need to combine a decision tree with a bagging classifier because you can easily use the classifier-class of random forest.

Random forest adds additional randomness to the model, while growing the trees. Instead of searching for the most important feature while splitting a node, it searches for the best feature among a random subset of features. This results in a wide diversity that generally results in a better model.

Therefore, in random forest, only a random subset of the features is taken into consideration by the algorithm for splitting a node. You can even make trees more random by additionally using random thresholds for each feature rather than searching for the best possible thresholds.

Feature Importance:

One of the best qualities of the random forest algorithm is that it is very easy to measure the relative importance of each feature on the prediction. By looking at the feature importance you can decide which features to possibly drop because they don't contribute enough (or sometimes nothing at all) to the prediction process. This is important because a general rule in machine learning is that the more features you have the more likely your model will suffer from overfitting and vice versa [5].

Merits and Demerits of Random forest Algorithm:

- **Versatility:** It can be used for both regression and classification tasks and it's also easy to view the relative importance it assigns to the input features.
- **Handy:** Random forest is also a very handy algorithm because the default hyper parameters it uses often produce a good prediction result. Understanding the hyper parameters is pretty straightforward, and there's also not that many of them.
- **No overfitting of data:** If there are enough trees in the forest, the classifier won't over fit the model.

Demerits include:

- The main limitation of random forest is that a large number of trees can make the algorithm too slow and ineffective for real-time predictions. In general, these algorithms are fast to train, but quite slow to create predictions once they are trained.
- A more accurate prediction requires more trees, which results in a slower model.
- In most real-world applications, the random forest algorithm is fast enough but there can certainly be situations where run-time performance is important and other approaches would be preferred.
- random forest is a predictive modelling tool and not a descriptive tool, meaning if you're looking for a description of the relationships in your data, other approaches would be better

Random forests are also very hard to beat performance wise

4. RESULTS AND DISCUSSIONS

The machine learning algorithms especially the classifiers are evaluated based on their performance. Classifiers often report a set of performance parameters such as follows:

The following are the variables defined which are used in the performance parameters.

TP = True Positive, TN = True Negative

FP = False Positive (false alarm, Type I error)

FN = False Negative (a miss, Type II error)

P = Number of Real Positives in the data

N = Number of Real Negatives in the data

Accuracy: The accuracy of a classifier is

$$a = \frac{TP + TN}{P + N} = \frac{TP + TN}{(TP + FN) + (TN + FP)}$$

Confusion Matrix: A confusion matrix is a table that summarizes the performance of the classifier as follows:

Table 2: Representing confusion matrix

n=100	Predicted Negative	Predicted Positive	
Actual Negative	TN = 40	FP = 10	TN + FP = N = 50
Actual Positive	FN = 5	TP = 45	FN + TP = P = 50
	TN + FN = 45	FP + TP = 55	

Error rate of misclassification rate is $1 - a$. The specificity is $1 - FP$, and it is equivalent to the true negative rate.

Precision p of classification for each class is a measure of its prediction accuracy for that class. Hence, for the positive class, it is the number of correct positives divided by the number of all positives predicted for that class, even if some of them are false positives. That is, p is

$$p = \frac{TP}{TP + FP}$$

Precision, therefore, provides a relative comparison of the classifier performance among the different classes. That is, the classifier be an unbalanced performer by doing better at prediction for one class versus another class.

Recall r is the number of correct positive results divided by the number of positive results that should have been returned.

$$r = \frac{TP}{P} = \frac{TP}{(TP + FN)}$$

The F1 score is the harmonic mean of precision and recall such that

$$F1 = 2 \times \frac{p \cdot r}{p + r}$$

It is interpreted as a weighted average of the precision and recall (best is 1 and worst is 0). Note that F1 is 1 if $p = r = 1$ and zero if either p or r is zero.

$$F1 = \frac{2 \cdot TP}{2 \cdot TP + FP + FN}$$

ROC Curve

ROC stands for Receiver Operating Characteristic. It is a graphical plot of True Positive Rate (sensitivity) from the positive samples versus False Positive Rate (false alarm) from the negative samples, as a function of a parameter of the classifier. Often, the parameter is the location of the decision boundary. The sensitivity is plotted on the y-axis and the false alarm is plotted on the x axis.

The whole dataset is split into Training dataset and Test dataset in 80:20 ratio respectively.

The hyper parameters are tuned according to the machine learning model and the machine learning model is trained with tuned hyper parameters. The SVC model is tuned by Max iterations , regularization constant , The Logistic regression model is tuned with Maximum iterations and regularization constant and the random forest model is tuned with number of trees and maximum depth of the tree .

The data set is preprocessed to obtain two different kinds of datasets.

1. Outliers from each of the class and resulting in a dataset without outliers.

2. The new dataset is obtained by selecting features using mRMR (minimum Redundancy Maximum Relevance) algorithm.

Performance parameters obtained from the experiments are recorded for evaluating the machine learning models and for comparison. The model is trained on both the type of datasets mentioned above.

The results obtained from the SVC model are as follows:

Table 3 shows the results when the SVC model was trained and tested with dataset from which outliers were removed.

Table 3: Performance parameters of SVC model containing dataset without outliers

Class	Precision	Recall	F1-Score
N(Normal)	0.984	0.822	0.896
S(Suspect)	0.427	0.814	0.56
P(Pathologic)	0.729	0.946	0.824
Weighted average	0.897	0.833	0.851

The below table 4 shows the results when the SVC model was trained and tested with dataset containing selected features.

Table 4: Performance parameters of SVC model containing dataset with reduced feature set

Class	Precision	Recall	F1-Score
N(Normal)	0.992	0.832	0.905
S(Suspect)	0.48	0.837	0.61
P(Pathologic)	0.63	0.919	0.747
Weighted average	0.9	0.841	0.857

The results obtained from the logistic regression model are as follows:

Table 5 shows the results when the logistic regression model was trained and tested with dataset from which outliers were removed.

Table 5: Performance parameters of logistic regression model containing dataset without outliers

Class	Precision	Recall	F1-Score
N(Normal)	.980	.789	.874
S(Suspect)	.391	.791	.523
P(Pathologic)	.673	.946	.787
Weighted average	.884	.804	.826

The below table 6 shows the results when the logistic regression model was trained and tested with dataset containing selected features.

Table 6: Performance parameters of logistic regression model containing dataset with reduced feature set

Class	Precision	Recall	F1-Score
N(Normal)	.976	.802	.880
S(Suspect)	.407	.767	.532
P(Pathologic)	.660	.946	.778
Weighted average	.882	.812	.831

The results obtained from the Random forest model are as follows:

Table 7 shows the results when the Random forest model was trained and tested with dataset from which outliers were removed.

Table 7: Performance parameters of Random forest model containing dataset without outliers

Class	Precision	Recall	F1-Score
N(Normal)	.980	.970	.975
S(Suspect)	.760	.884	.817
P(Pathologic)	1.000	.892	.943
Weighted average	.957	.953	.954

The below table 8 shows the results when the Random forest model was trained and tested with dataset containing selected features.

Table 8: Performance parameters of Random forest model containing dataset with reduced feature set

Class	Precision	Recall	F1-Score
N(Normal)	.974	.977	.975
S(Suspect)	.800	.837	.818
P(Pathologic)	.971	.892	.930
Weighted average	.954	.953	.953

The comparison of performance parameter Accuracy of all the models is depicted in the table 9 below.

Table 9: comparison of accuracies across different algorithms

Model	Dataset without Outliers	Dataset with reduced features
SVC	0.833	0.841
Logistic regression	0.804	0.812
Random forest	0.953	0.953

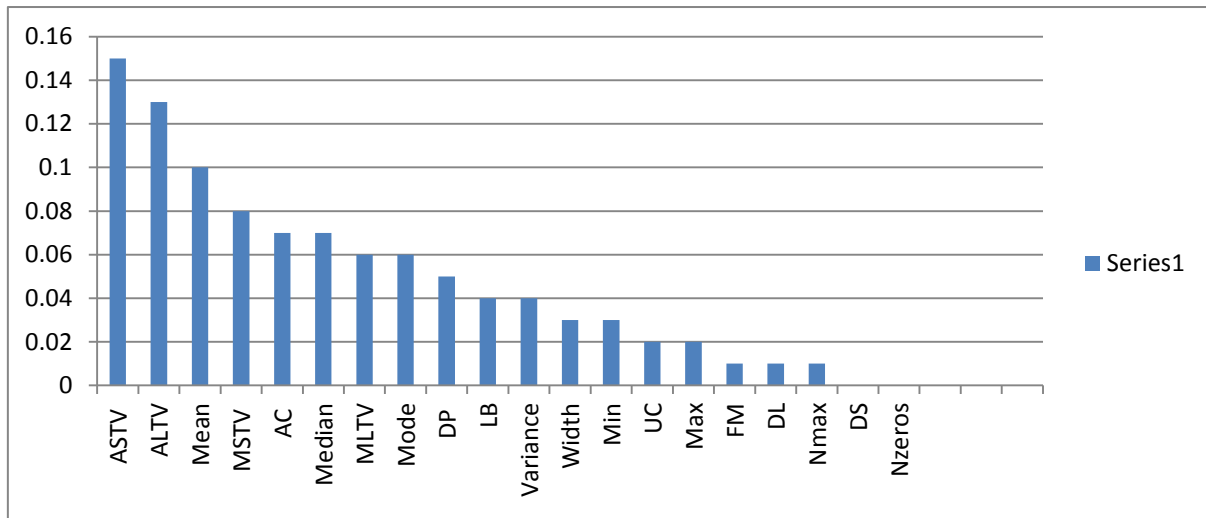
It is evident from the table that Random forest analysis prediction has the highest accuracy of .953 or 95.3%.

The results are compared with results published in related work by [6] Muhammad Arif in paper "Classification of cardiocograms using random forest classifier and selection of important features from cardiocogram signal"

Table 10: comparison of results with previous work

Reference	Method	Results
Muhammad Arif 2015[6]	Random Forest (Full Features)	Overall accuracy 93.6%
Current paper	Random forest wit hyper parameter tuning and outlier removal technique	Accuracy achieved 95.3%

The following graph shows the variable/feature importance.



5. CONCLUSION

In this paper we evaluated the performance of Support vector machine (SVM), Logistic regression and random forest classification algorithm based models for predicting the fetal health using cardiotocography data. The models were trained with two different kinds of preprocessed data sets one with outliers removed in it and other with selected features in it. The models were tuned with learned hyper parameters of the model. The performance indicators measured were namely Precision, Recall and F1 scores. It was observed that accuracy achieved was highest for random forest classifier with 95.3 %. The dataset referred here is confined to a particular geographical area so this may be enhanced in future to include the data from across the globe to represent all the different kind of data.

REFERENCES

1. German Society of Gynecology and Obstetrics (DGGG), Maternal Fetal Medicine Study Group (AGMFM), German Society of Prenatal Medicine and Obstetrics (DGPGM), German Society of Perinatal Medicine (DGPM) "S1-Guideline on the Use of CTG During Pregnancy and Labor" Guideline 721
2. Ayres de Campos et al. (2000) SisPorto 2.0 A Program for Automated Analysis of Cardiotocograms. J Matern Fetal Med 5:311-318
3. Nello Cristianini and John Shawe-Taylor, "An Introduction to Support Vector Machines and Other Kernel-based Learning Methods", Cambridge University Press, 2000.
4. <https://scikit-learn.org/>
5. <https://builtin.com/data-science/random-forest-algorithm>.
6. Muhammad Arif, "Classification of cardiotocograms using random forest classifier and selection of important features from cardiotocogram signal", Biomaterials and Biomedical Engineering, Vol. 2, No. 3 (2015) 173-183
7. <https://pandas.pydata.org/>