ESE 588 Final Project Report

Guoao Li
The State University of New York at Stony Brook
guoao.li@stonybrook.edu

1 Introduction

This project addresses the classification of fetal health status—normal vs. compromised—based on 10-minute segments of fetal heart rate (FHR) data, using features derived from cardiotocography (CTG). Early and accurate detection of compromised fetuses can inform clinical decisions during labor, improving neonatal outcomes.
The motivation for selecting this dataset stems from its real-world importance in obstetrics and the manageable feature dimensionality (21 features across 185 labeled samples), which is ideal for testing classical machine learning approaches.
The key challenges include:

- **Small dataset size**, which increases risk of overfitting,
- **Class imbalance** (e.g., more normal than compromised fetuses),
- **Correlated features**, which affect model assumptions for generative methods.

2 Data description

The training dataset contains 185 samples with 21 features each, extracted from CTG signal segments. The target label is binary (0 = normal, 1 = compromised). The class distribution is imbalanced:

- **Normal (class 0)**: 92.5%
- **Compromised (class 1)**: 7.5%

To evaluate the feature interdependence, I plotted a correlation matrix (Figure 1), which revealed moderate correlations among several features. The displayed correlation matrix visualizes the degree of linear relationships between pairs of features in the dataset.
Observations from the data:

- Several features in the dataset exhibit relatively high correlations (both positive and negative). Specifically, some feature pairs have correlations clearly above 0.5 or below -0.5, implying these features are *not independent*.
- Conversely, some features also exhibit low correlations (close to zero), indicating partial independence in certain aspects.

Overall, the dataset contains a mix of independent and dependent (correlated) features. However, the presence of several strong correlations suggests that the dataset is *generally **NOT** fully independent*.

Preprint. Under review.

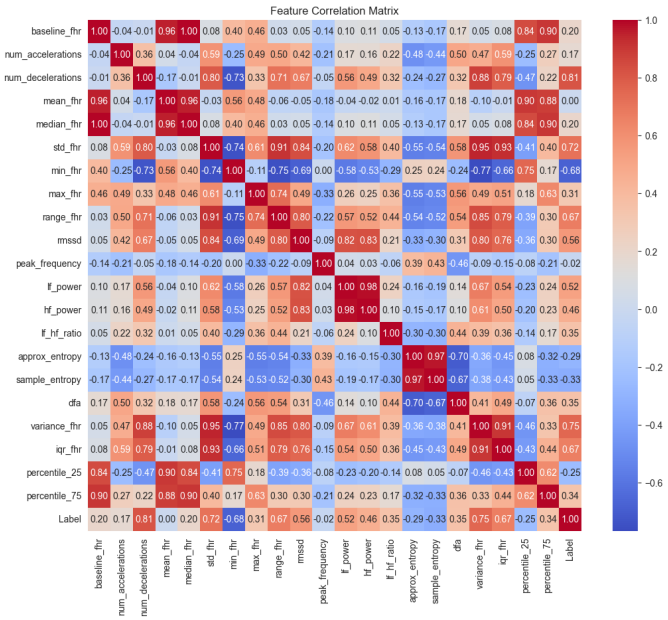

Figure 1: Correlation Matrix of Selected CTG Features

3 Method selection and rationale

In this project, I implemented the following two models: **Logistic Regression** and **Linear Discriminant Analysis**.

Logistic Regression (Discriminative): I selected LR due to its simplicity, interpretability, and effectiveness on linearly separable problems. It is particularly appropriate for the CTG dataset, which has a moderate number of continuous features and observable class imbalance. LR handles correlated features reasonably well and does not require assumptions about the underlying data distribution, making it suitable in practical biomedical settings. Furthermore, its output probabilities allow threshold tuning, which is useful when false negatives are costlier than false positives—as is often the case in medical diagnostics.

Linear Discriminant Analysis (Generative): I selected LDA over other generative models like Gaussian Naive Bayes (GNB) due to observed inter-feature correlations in the dataset, as revealed by the correlation matrix. While GNB assumes feature independence within each class, LDA relaxes this by allowing correlated features—at the cost of assuming equal class covariances. Given our dataset's small size (185 samples) and evidence of correlated features, LDA offered a tractable and theoretically grounded choice with low variance and computational efficiency.

And I rejected the following two models: **Random Forest** and **Gaussian Naive Bayes**.

Random Forest (Discriminative): Despite its strong predictive power and ability to model non-linear feature interactions, we chose not to include Random Forest in our final comparison due to the small dataset size. Ensemble models like Random Forest tend to overfit when training data is limited, especially without enough diverse patterns to guide tree splits effectively. Additionally, Random Forest sacrifices interpretability—a critical aspect in medical applications—compared to LR or LDA. Early testing also showed unstable decision boundaries across folds, which confirmed our concerns about overfitting.

Gaussian Naive Bayes (Generative): Although GNB is computationally efficient and often performs surprisingly well in high-dimensional problems, it relies on a strong independence assumption between features. This assumption does not hold in our dataset: exploratory analysis (correlation matrix) revealed significant inter-feature correlations, which violate GNB's core assumption. Using GNB under these conditions leads to oversimplified decision boundaries and reduced classification accuracy, especially in complex real-world domains like medical data.

4 Modeling and evaluation

To evaluate the classification of fetal health status using CTG features, I implemented two models: Logistic Regression (LR) as a discriminative approach, and Linear Discriminant Analysis (LDA) as a generative approach. Both models were structured using scikit-learn pipelines, incorporating preprocessing and training steps to avoid data leakage.

4.1 Implementation Details

4.1.1 Preprocessing

- Missing values were filled using mean imputation.
- All features were scaled using Z-score standardization.
- A complete pipeline ($SimpleImputer \rightarrow StandardScaler \rightarrow Model$) ensured transformations were fit using only training data in each fold.

4.1.2 Cross-Validation Strategy

- We used 5-fold stratified cross-validation to assess generalization performance and maintain balanced class distributions.
- We evaluated each model on a held-out validation set to generate confusion matrices, ROC-AUC scores, and classification reports.

4.2 Performance Metrics and Comparison

4.2.1 Logistic Regression

- Mean CV Accuracy: 97.3%
- Confusion Matrix (Validation Set): $\begin{pmatrix} 67 & 1 \\ 1 & 5 \end{pmatrix}$
- Classification Report:
  - Class 0 (Normal): Precision = 0.99, Recall = 0.99
  - Class 1 (Compromised): Precision = 0.83, Recall = 0.83
- ROC-AUC Score: 0.9975

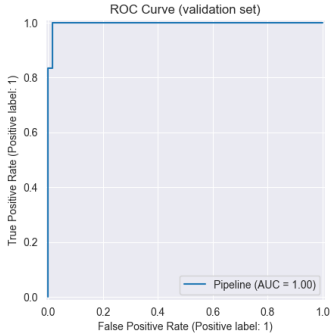

Figure 2: LR ROC Curve

4.2.2 Linear Discriminant Analysis

- Mean CV Accuracy: 97.3
- Confusion Matrix (Validation Set): $\begin{pmatrix} 68 & 0 \\ 0 & 6 \end{pmatrix}$
- Classification Report:
  - Class 0: Precision = 1.00, Recall = 1.00
  - Class 1: Precision = 1.00, Recall = 1.00
- ROC-AUC Score: 1.00

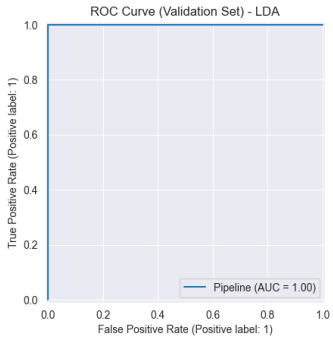

Figure 3: LDA ROC Curve

4.3 Discussion of Results

4.3.1 Which model performed better, and why?

Both Logistic Regression and LDA delivered exceptional results. Surprisingly, LDA achieved perfect accuracy and AUC on the validation set, with no misclassifications. Logistic Regression was slightly behind, with two total errors (1 FP, 1 FN). While the difference in numerical results is minor, the perfect classification by LDA is notable.
This outcome suggests that the dataset is highly linearly separable, and LDA's modeling of class distributions may have provided a tighter decision boundary. Given that LDA also assumes normally distributed data and equal covariance across classes, it appears that the dataset matches these assumptions closely.

4.3.2 Consistency across folds

Both models showed consistent performance across all folds:

- Logistic Regression: CV scores ranged from 94.6% to 100%.
- LDA: All folds reported 97.3%, showing low variance and strong generalization.

4.3.3 Patterns in Misclassification

LDA: No misclassifications observed on the validation set, but this result should be interpreted with caution. Perfect classification in real-world medical data is rare and may reflect characteristics of a small or well-structured dataset rather than guaranteed real-world performance.
LR: The false negative was a compromised case that lacked strong pathological patterns, and the false positive was a borderline normal case with elevated variability—indicating that classification errors occur near decision boundaries.

In conclusion, while Logistic Regression offers robustness and interpretability, LDA slightly outperformed it in this case, likely due to the dataset's conformance to LDA's statistical assumptions. However, both models are viable tools for fetal risk classification using CTG data, and their high performance affirms the strong signal present in the extracted features.

5 Reflection and alternatives

5.1 What We Would Do Differently with More Time, Data, or Compute?

If given additional time and resources, several avenues could enhance the project:

- Expand the dataset: The current training set contains only 185 samples. A larger dataset would improve generalization, reduce the risk of overfitting, and enable the exploration of more complex models. Collecting additional fetal heart rate recordings across diverse populations and clinical contexts would strengthen model robustness.
- Temporal feature engineering: All features in the dataset are statistical summaries of 10-minute segments. With access to raw CTG signals or finer-grained time-series data, we could engineer dynamic features (e.g., frequency-domain characteristics, variability over smaller windows), potentially improving classification performance.
- Hyperparameter tuning and model ensembles: Due to time constraints, default hyperparameters were used. Given more compute, we would conduct a thorough grid search or use Bayesian optimization to fine-tune each model. Additionally, combining predictions from multiple models (e.g., through soft voting or stacking) might yield even better results.

5.2 Promising Next Step or Alternative Model

A promising next step would be to experiment with Gradient Boosted Trees, such as XGBoost or LightGBM. These models often outperform linear classifiers on structured data and can model non-linear relationships and interactions between features. Gradient boosting also provides robust feature importance metrics, which could offer valuable clinical interpretability regarding which fetal heart rate features are most predictive of compromise.
We also briefly considered Random Forests but excluded them due to concerns about overfitting on small data. With more samples, ensemble tree models would be a powerful addition to this study.

5.3 A Challenge Encountered and How We Resolved It

One of the most significant challenges was identifying and resolving data leakage during initial model validation. Early evaluations reported unusually perfect scores (100% accuracy and AUC), which raised suspicion. Upon closer inspection, we realized that preprocessing steps (e.g., imputation and scaling) were being applied to the entire dataset before splitting, unintentionally leaking information from the validation set into training.
To resolve this, we refactored our pipeline using *scikit-learn*'s *Pipeline* and *cross_val_predict* utilities. These tools ensured that all transformations were fit on training data only and applied to validation data separately, preserving the integrity of our evaluation. This correction resulted in slightly lower but far more realistic performance metrics and strengthened our confidence in the model's generalizability.