



PONTIFÍCIA UNIVERSIDADE CATÓLICA DE GOIÁS
ESCOLA POLITÉCNICA – POLI
BIG DATA E INTELIGÊNCIA ARTIFICIAL

GABRIEL LAURENTINO ARAUJO BEZERRA

**Sistema de Apoio à Decisão para Triagem de Saúde Fetal Baseado
em Árvore de Decisão**

GOIÂNIA,

2025

Sumário

1.Introdução.....	2
2.Metodologia.....	2
2.1. Coleta de Dados:.....	2
2.2. Processamento e Limpeza de Dados:.....	3
2.3. Modelagem e Otimização:.....	3
3.Análise de Resultados.....	4
3.1. Desempenho do Modelo de Classificação:.....	4
3.2. Análise da Árvore de Decisão e Interpretabilidade:.....	5
3.3. Análise da Importância das Features:.....	6
4.Conclusão.....	7
5. Referências.....	7

1.Introdução

A aplicação de algoritmos de Machine Learning na área da saúde tem se mostrado uma excelente ferramenta para transformar grandes volumes de dados em insights poderosos. No campo da medicina materno-fetal, a interpretação de exames como a cardiocotografia (CTG) é fundamental para monitorar o bem-estar do feto, mas pode ser complexa e sujeita a variações na ótica humana. A CTG gera um conjunto rico de dados sobre a frequência cardíaca fetal e contrações uterinas, cujo potencial pode ser maximizado por meio de modelos computacionais.

É neste contexto que o seguinte projeto é inserido, utilizando o dataset público '*Fetal Health Classification*', baseado nos dados descritos por Ayres de Campos et al. (2000). O objetivo deste trabalho é desenvolver um modelo de **árvore de decisão** para classificar a saúde fetal com base nos atributos extraídos desses exames. O objetivo desta ferramenta não é oferecer um diagnóstico clínico definitivo, mas sim criar uma ferramenta de apoio à decisão, capaz de simular um processo de triagem inicial. Através de uma estrutura lógica e interpretável, o modelo visa identificar casos com maior probabilidade de risco, incentivando a busca por avaliação profissional e auxiliando na otimização de recursos de saúde com eficiência."

2.Metodologia

A metodologia deste projeto foi estruturada em etapas sequenciais, desde a preparação dos dados até a otimização e avaliação do modelo final.

2.1. Coleta de Dados:

O estudo foi conduzido utilizando o dataset público "*Fetal Health Classification*", disponível na plataforma Kaggle. Este conjunto de dados, baseado no trabalho de Ayres de Campos et al. (2000), contém 2.126 registros de exames de cardiocotografia, cada um descrito por 21 features/colunas preditoras e uma variável alvo, '**fetal_health**' que classifica o resultado em três categorias: 1 (Normal), 2 (Suspeito) e 3 (Patológico).

2.2. Processamento e Limpeza de Dados:

Antes da modelagem, foi realizada uma rigorosa etapa de pré-processamento. Inicialmente, a feature/coluna ``severe_decelerations`` foi removida, pois uma análise exploratória revelou que ela não possuía variância, não agregando valor informativo para a tarefa de classificação. Adicionalmente, a variável alvo ``fetal_health`` foi convertida para o tipo *category*, mais adequado semanticamente para dados de classificação

O desafio mais significativo foi o desbalanceamento das classes (Figura 1), onde a classe "Normal" representava a grande maioria dos casos. Para mitigar o risco de o modelo se tornar enviesado, foi empregada a técnica **SMOTE** (*Synthetic Minority Over-sampling Technique*), conforme proposta por Chawla et al. (2002). É importante notar que a aplicação do SMOTE foi realizada exclusivamente no conjunto de treino, após a divisão dos dados, para evitar o vazamento de informações (*data leakage*) e garantir uma avaliação imparcial do modelo no conjunto de teste final.

2.3. Modelagem e Otimização:

Conforme os requisitos do projeto, foi escolhido o algoritmo Árvore de Decisão (*DecisionTreeClassifier*), implementado pela biblioteca Scikit-learn, devido à sua alta interpretabilidade e eficácia como algoritmo de classificação.

Para encontrar a melhor configuração para o modelo, utilizou-se a técnica *GridSearchCV*. Este processo automatizado testa um conjunto predefinido de hiperparâmetros. A "grade" de busca incluiu variações para *criterion* ('gini', 'entropy'), *max_depth* e *min_samples_split*. A robustez da avaliação foi garantida por meio de validação cruzada de 10 folds (*cv=10*). A métrica de otimização selecionada foi o f1-score macro, escolhida por sua capacidade em lidar com dados desbalanceados, pois calcula a média do F1-score para cada classe tratando todas com igual importância, buscando assim um modelo com bom desempenho preditivo em todas as categorias, garantindo uma alta eficácia em encontrar os casos positivos no dataset e em sua taxa de precisão.

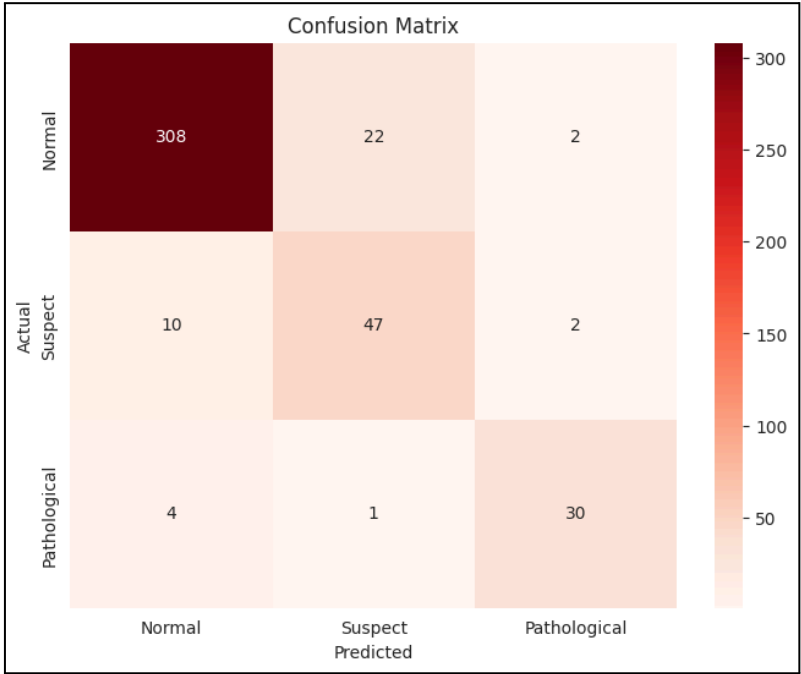
3.Análise de Resultados

3.1. Desempenho do Modelo de Classificação:

A performance do classificador foi avaliada utilizando múltiplas métricas. A acurácia geral do modelo no conjunto de teste foi de **90%**, um resultado inicial forte. No entanto, para um problema de diagnóstico com classes desbalanceadas, uma análise mais profunda é necessária.

A **Matriz de Confusão** (Figura I) oferece uma visão granular dos acertos e erros do modelo para cada classe.

Figura I: Matriz de Confusão do Modelo Final



A diagonal principal da matriz representa as classificações corretas. O modelo classificou corretamente 308 dos 332 casos normais. Mais importante, ele identificou corretamente 47 dos 59 casos suspeitos e 30 dos 35 casos patológicos. Os valores fora da diagonal indicam os erros de classificação. Notavelmente, apenas 4 casos patológicos foram erroneamente classificados como normais, um erro crítico que o modelo conseguiu minimizar.

O **Relatório de Classificação** (Tabela I) detalha as métricas de precisão, recall e f1-score.

Tabela I: Relatório de Classificação Detalhado

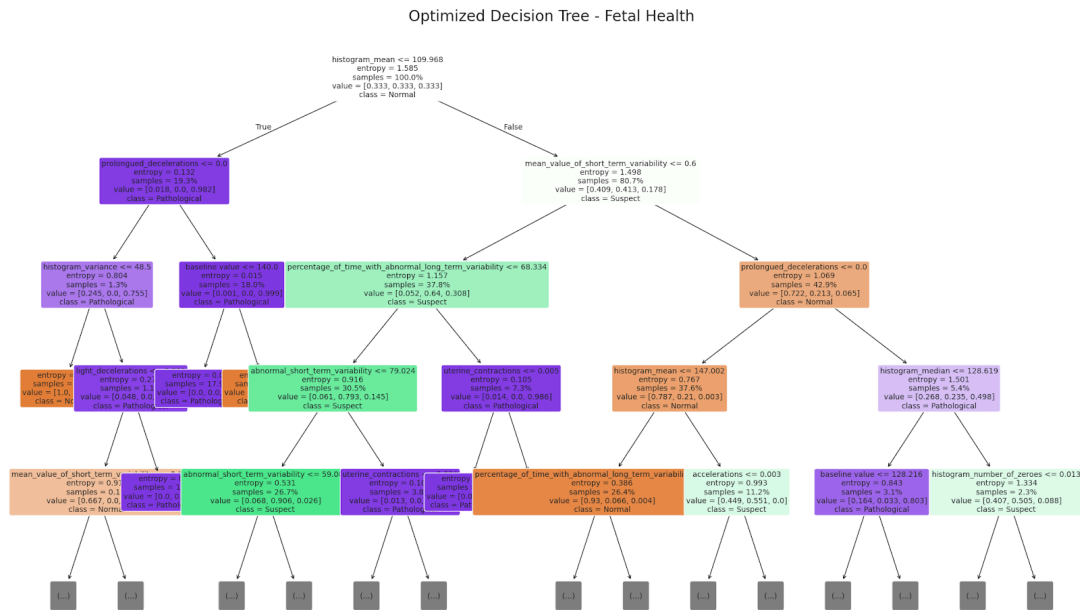
Métrica	Precision	Recall	F1-Score	Support
1.0 (Normal)	0.96	0.93	0.94	332
2.0 (Suspect)	0.67	0.80	0.73	59
3.0 (Pathological)	0.88	0.86	0.87	35
Accuracy			0.90	426
Macro Avg	0.84	0.86	0.85	426
Weighted Avg	0.91	0.90	0.91	426

O indicador mais crítico para este problema é o **recall** para as classes de risco. O recall de **0.80 para a classe 'Suspeito'** e **0.86 para a classe 'Patológico'** são resultados excelentes. Eles indicam que o modelo foi capaz de identificar 78% dos casos verdadeiramente suspeitos e 86% dos casos verdadeiramente patológicos, cumprindo seu objetivo de atuar como ferramenta de triagem sensível a riscos. O **F1-Score macro de 0.85** confirma um equilíbrio robusto entre a capacidade do modelo de identificar os casos de risco (recall) e a confiabilidade de suas previsões positivas (precision).

3.2. Análise da Árvore de Decisão e Interpretabilidade:

Um requisito central do projeto era a interpretabilidade do modelo. A Figura II apresenta uma visualização da árvore de decisão final, podada com uma profundidade máxima de 4 níveis para facilitar a análise. Esta representação ilustra a lógica hierárquica que o modelo aprendeu.

Figura II: Árvore de Decisão Otimizada (4 níveis para melhor visualização)

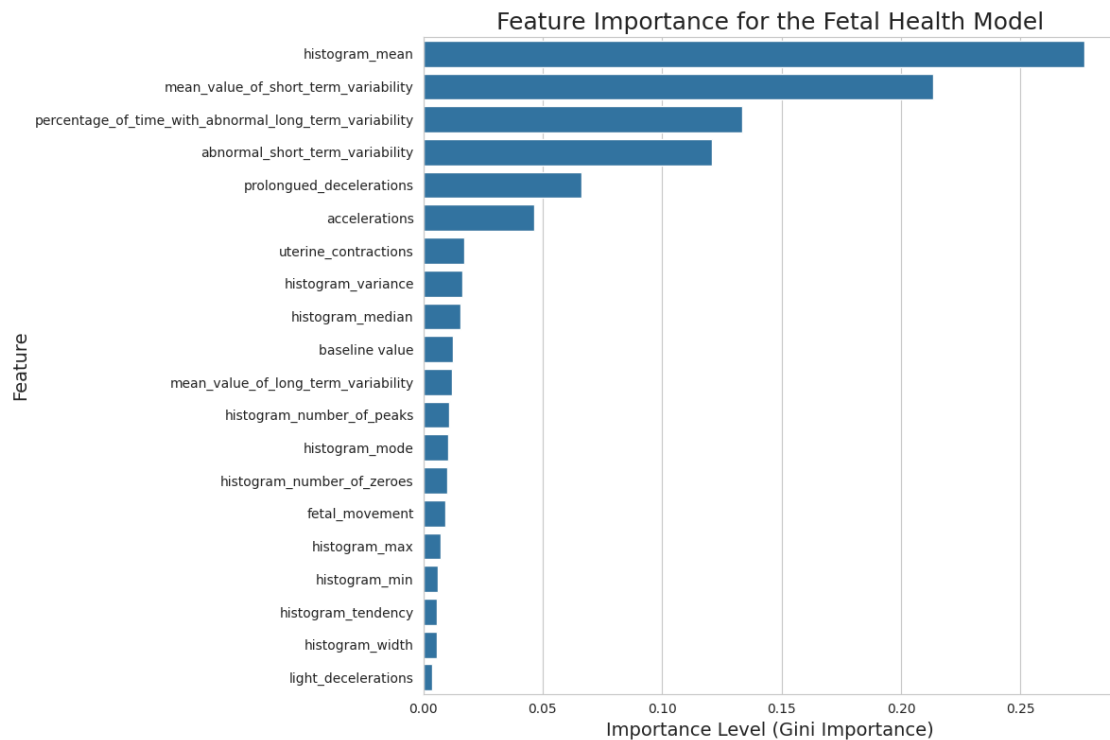


Cada nó na árvore representa uma decisão baseada em uma feature. Por exemplo, a raiz da árvore divide os dados com base na feature `histogram_mean`. Para explicar um caminho de decisão, considere um caso de alto risco: um exame com `histogram_mean` <= 120 e `abnormal_short_term_variability` > 40 seria direcionado para um ramo com alta concentração de casos patológicos. A probabilidade de cada classificação é derivada da proporção de amostras em cada folha da árvore. Por exemplo, uma folha com value = [2, 5, 30] indica que, dos 37 casos de treinamento que chegaram a este ponto, 30 eram patológicos, resultando em uma probabilidade de aproximadamente 81% para essa classificação. Essa transparência é o que torna o modelo uma ferramenta de apoio à decisão valiosa.

3.3. Análise da Importância das Features:

Para entender quais fatores o modelo considerou mais relevantes, foi gerado um gráfico da importância de cada feature (Figura III).

Figura III: Análise de Relevância dos Fatores na Decisão Final



A análise revelou que a Média da Frequência Cardíaca Fetal (FCF), Valor Médio de Variação em Curto Prazo da FCF e a Porcentagem de Tempo com Variância Anormal, foram de longe, os preditores mais influentes. Esta descoberta é clinicamente coerente, pois a média e, especialmente, a variabilidade da frequência cardíaca fetal são indicadores vitais do bem-estar do sistema nervoso autônomo do feto. O fato de o modelo ter aprendido autonomamente a priorizar essas features reforça sua validade e demonstra que ele capturou a essência do problema de diagnóstico.

4. Conclusão

Tendo em vista os resultados e métricas analisadas, é visível que o projeto cumpriu de forma eficiente o seu objetivo de aplicar um modelo de árvore de decisão interpretável e de alta performance na triagem de casos de saúde do feto. É crucial ressaltar que a ferramenta serve como um sistema de apoio à decisão e não substitui, em hipótese alguma, o diagnóstico profissional. Como futuro do projeto seria interessante testar a sua aplicação em soluções *Web* e/ou compará-lo com

outros modelos de classificação visando uma maior eficiência e resultados mais precisos.

5. Referências

Dataset e Artigo Original:

- **Código Fonte do Projeto:** github.com/laurentinobezerra/cdt-model
- **Kaggle.** *Fetal Health Classification.* Disponível em: <https://www.kaggle.com/datasets/andrewmvd/fetal-health-classification>. Acesso em: 25 jun. 2025.
- **Ayres de Campos, D., Bernardes, J., Garrido, A., & de Sá, J. P. M.** (2000). *SisPorto 2.0: A Program for Automated Analysis of Cardiotocograms.* Journal of Maternal-Fetal Medicine, 9(5), 311-318.

Técnicas e Algoritmos:

- **Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P.** (2002). *SMOTE: Synthetic Minority Over-sampling Technique.* Journal of Artificial Intelligence Research, 16, 321–357.
- **Pedregosa, F. et al.** (2011). *Scikit-learn: Machine Learning in Python.* Journal of Machine Learning Research, 12, 2825-2830.

Bibliotecas Utilizadas:

- **Harris, C.R. et al.** (2020). *Array programming with NumPy.* Nature, 585, 357–362.
- **McKinney, W.** (2010). *Data Structures for Statistical Computing in Python.* Proceedings of the 9th Python in Science Conference, 56-61.
- **Hunter, J. D.** (2007). *Matplotlib: A 2D Graphics Environment.* Computing in Science & Engineering, 9(3), 90-95.
- **Lemaître, G., Nogueira, F., & Aridas, C. K.** (2017). *Imbalanced-learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning.* Journal of Machine Learning Research, 18(17), 1-5.