UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE CIÊNCIA DA COMPUTAÇÃO

TAIANE DE OLIVEIRA PEIXOTO

Comparação de estratégias para lidar com o desbalanceamento de classes: um estudo de caso com dados de mortalidade neonatal no Rio Grande do Sul

Monografia apresentada como requisito parcial para a obtenção do grau de Bacharel em Ciência da Computação

Orientador: Prof. Dr. Mariana Recamonde Mendoza
Co-orientador: Profa. Dra. Thayne Woycinck Kowalski

Porto Alegre
2023

“Eu sonho mais alto que drones...“ —(EMICIDA, 2019)

AGRADECIMENTOS

Agradeço aos meus pais, por fazerem o possível e o impossível para garantir que eu tivesse os melhores meios possíveis de acesso à educação. A minha mãe, Janaína, agradeço por sempre me incentivar a estudar e aproveitar as oportunidades que ela não teve. Ao meu pai, Márcio, por sempre reafirmar que eu poderia fazer o que eu quisesse.

Agradeço aos meus irmãos que me apoiam e enxergam a mim como exemplo a ser seguido, sempre tento ser uma pessoa melhor por causa deles.

Agradeço à minha orientadora, Mariana, por ser um excelente professora e pesquisadora da área de aprendizado de máquina e inspiração para muitas pessoas que querem seguir nessa área, agradeço pela orientação cuidadosa nesse trabalho nos últimos meses.

Agradeço minha terapeuta, Carol, que me acompanha nos últimos meses e me ajuda a conseguir lidar com meus desafios diários, mesmo aqueles que são pequenos para o mundo mas enormes para mim.

Agradeço aos meus amigos, muitos eu conheci na universidade ou por causa da universidade, amigos que me acompanham há muitos anos, outros conheci recentemente, agradeço todos vocês por estarem presentes nos momentos alegres e também difíceis dessa jornada que é a graduação.

Agradeço todos os professores que se dedicam diariamente ao ensino e pesquisa mesmo com todos os desafios que existem no ensino superior público do Brasil.

Agradeço também as ações afirmativas que buscam tornar a universidade a cada dia mais acessível para todas as pessoas.

Finalmente, agradeço a todas as pessoas que passaram pela minha jornada, ou que de alguma forma fazem parte da construção do curso de Ciência da Computação da UFRGS, esse trabalho de conclusão representa apenas uma parcela do que foi todo o caminho percorrido durante a minha graduação.

RESUMO

A Taxa de Mortalidade Infantil (TMI) é considerada um dos indicadores mais relevantes das condições de vida de uma população. No ano de 2020, a TMI foi de 8,62/1000 nascidos vivos (NV) no estado do Rio Grande do Sul (RS), atingindo a meta anual firmada pelo estado de 9,75/1000. Em torno de 77,49% dos casos foram óbitos neonatais, isto é, antes de 28 dias de vida completos. Tendo em vista que saúde é um dos indicadores brasileiros para os objetivos de desenvolvimento sustentável e esse objetivo inclui a meta de reduzir a Taxa de Mortalidade Neonatal (TMN), é importante identificar os fatores associados com a TMN no Brasil e suas regiões, e investigar a utilização dos mesmos para o treinamento de modelos preditivos para o risco de óbito neonatal aplicando, por exemplo, Aprendizado de Máquina (AM). Visto que esta tarefa de classificação lida com uma distribuição de classes inerentemente desbalanceada, torna-se necessário investigar o impacto do desbalanceamento de classes no desempenho de algoritmos e a efetividade de estratégias existentes para lidar com este desafio. Assim, este trabalho analisa estratégias computacionais para lidar com o desbalanceamento de classes em AM em dados de óbito neonatal do RS. Foram avaliados quatro algoritmos de classificação baseados em árvores de decisão e seis métodos para lidar com o desbalanceamento de classes, incluindo métodos de amostragem, métodos baseados em modificações de algoritmos ensemble e uma abordagem sensível ao custo. Ao final, os desempenhos dos modelos preditivos foram comparados e avaliados para uma base de dados construída a partir do pré-processamento e integração dos dados do Sistema de Informação sobre Nascidos Vivos (SINASC) e Sistema de Informação sobre Mortalidade (SIM) para o RS, apresentando 99.6% de instâncias na classe negativa. O classificador XGBoost combinado com o método SMOTE-ENN foi o que melhor lidou com o desbalanceamento de classes nesse domínio, alcançando 73% de acurácia balanceada, 46% de sensibilidade e 46% de score F1. Também foi constatado que o método SMOTE-ENN melhorou o desempenho dos modelos que utilizaram algoritmos de *boosting*, onde a sensibilidade aumentou em 8% no modelo com AdaBoost e 9% no modelo com XGBoost. Por fim, a abordagem sensível ao custo melhorou o desempenho dos modelos com árvore de decisão e florestas aleatórias, aumentando a sensibilidade em 26% no modelo com árvore de decisão e 45% no modelo com florestas aleatórias.

Palavras-chave: Aprendizado de máquina. mortalidade neonatal. desbalanceamento de classes.

Comparison of strategies to deal with class imbalance: a case study with neonatal mortality data in Rio Grande do Sul

ABSTRACT

The Infant Mortality Rate (IMR) is considered one of the most relevant indicators of the living conditions of a population. In 2020, the IMR was 8.62/1000 live births in the state of Rio Grande do Sul (RS), reaching the annual target set by the state of 9.75/1000. Around 77.49% of cases were neonatal deaths, that is, before 28 full days of life. Considering that health is one of the Brazilian indicators for sustainable development goals and this goal includes the target of reducing the Neonatal Mortality Rate (NMR), it is important to identify the factors associated with NMR in Brazil and its regions, and to investigate their use in training predictive models for the risk of neonatal death using, for example, Machine Learning (ML). Since this classification task deals with an inherently imbalanced class distribution, it is necessary to investigate the impact of class imbalance on algorithm performance and the effectiveness of existing strategies to deal with this challenge. Thus, this work analyzes computational strategies to deal with class imbalance in ML on neonatal death data in RS. Four classification algorithms based on decision trees and six methods for dealing with class imbalance were evaluated, including sampling methods, methods based on modifications of ensemble algorithms, and a cost-sensitive approach. Finally, the predictive model performances were compared and evaluated for a database constructed from the preprocessing and integration of data from the Live Birth Information System (SINASC) and Mortality Information System (SIM) for RS, presenting 99.6% of instances in the negative class. The XGBoost classifier combined with the SMOTE-ENN method was the one that best dealt with class imbalance in this domain, achieving 73% balanced accuracy, 46% sensitivity, and 46% F1 score. It was also found that the SMOTE-ENN method improved the performance of models that used boosting algorithms, where sensitivity increased by 8% in the AdaBoost model and 9% in the XGBoost model. Finally, the cost-sensitive approach improved the performance of models with decision trees and random forests, increasing sensitivity by 26% in the decision tree model and 45% in the random forests model.

Keywords: machine learning, neonatal mortality, class imbalance.

LISTA DE FIGURAS

Figura 2.1 Matriz de Confusão para um problema binário 22
Figura 2.2 Exemplo de uma curva ROC 24
Figura 2.3 Exemplo de uma curva precisão-recall 25
Figura 2.4 validação cruzada 5-fold 26

Figura 4.1 Relação de valores ausentes para cada atributo do conjunto de dados de treinamento 37
Figura 4.2 Relação de valores ausentes para cada atributo do conjunto de dados de teste 38
Figura 4.3 Distribuição das classes no conjunto de dados 39

Figura 5.1 Curvas de ROC e de *precision-recall* do modelo que utilizou o classificador XGBoost e método de amostragem SMOTE-ENN com dados de treino 42
Figura 5.2 Curvas de ROC e de *precision-recall* do modelo que utilizou o classificador XGBoost e método de amostragem SMOTE-ENN com dados de teste 43
Figura 5.3 Influência dos atributos nos modelos 45

LISTA DE TABELAS

Tabela 4.1 Porcentagem de valores ausentes nos campos utilizados para realizar a união de registros entre as bases *SINASC* e *SIM* 34
Tabela 4.2 Porcentagem de *linkages* realizados com sucesso entre as bases *SINASC* e *SIM* 35
Tabela 4.3 Grid Search e suas opções de hiperparâmetros 40
Tabela 4.4 Modelos e seus hiperparâmetros 40

Tabela 5.1 Resultados do experimento 44

LISTA DE ABREVIATURAS E SIGLAS

AD	Árvore de Decisão
AdaBoost	*Adaptive Boosting*
ADASYN	*Adaptive Synthetic*
AM	Aprendizado de Máquina
AUC	*Area Under the Curve*
AUPRC	Área sob a curva de *recall* e precisão
DATASUS	Departamento de informática do Sistema Único de Saúde
DNV	Declaração de Nascido Vivo
FA	Florestas Aleatórias
FN	Falso negativo (do inglês, *False negatives*)
FP	Falso positivo (do inglês, *False positives*)
IA	Inteligência Artificial
KNN	*K-Nearest Neighbors*
LGPD	Lei Geral de Proteção de Dados
NB	Naive Bayes
ON	Óbito Neonatal
RB	Redes Bayesianas
RL	Regressão Logística
RN	Redes Neurais
RNM	Redes Neurais Multicamadas
ROC	*Receiver Operating Characteristic*
RUSBoost	*Random Under Sampling Boosting*
SHAP	*SHapley Additive exPlanations*
SIM	Sistema de Informação sobre Mortalidade

SINASC	Sistema de Informação sobre Nascidos Vivos
SMOTE	*Synthetic Minority Oversampling Technique*
SMOTE-ENN	SMOTE com *Edited Nearest Neighbour*
SMOTE-Tomek	SMOTE com *Tomek links*
SVM	*Support Vector Machines*
TB	Teorema de Bayes
TN	Verdadeiro negativo (do inglês, *True negatives*)
TP	Verdadeiro positivo (do inglês, *True positives*)
TMI	Taxa de Mortalidade Infantil
TMN	Taxa de Mortalidade Neonatal
XGBoost	*Extreme Gradient Boosting Trees*

SUMÁRIO

1 INTRODUÇÃO 12
2 FUNDAMENTAÇÃO TEÓRICA 14
2.1 Óbito Neonatal 14
2.2 Métodos de Aprendizado de Máquina 14
2.2.1 Árvores de Decisão 15
2.2.2 Florestas Aleatórias 16
2.2.3 Adaptive Boosting 16
2.2.4 XGBoost 17
2.3 Desbalanceamento de Classes 17
2.3.1 Métodos de Amostragem 17
2.3.1.1 Subamostragem Aleatória 17
2.3.1.2 SMOTE 18
2.3.1.3 SMOTE-ENN 18
2.3.1.4 SMOTE-Tomek 19
2.3.2 Modificações de Algoritmos 19
2.3.2.1 Floresta Aleatória Balanceada 19
2.3.2.2 RUSBoost 20
2.3.3 Abordagem Sensível ao Custo com Floresta Aleatória Ponderada 20
2.4 Avaliação de Modelos de Aprendizado de Máquina 20
2.4.1 Matriz de Confusão 21
2.4.2 Acurácia 22
2.4.3 Acurácia Balanceada 22
2.4.4 Precisão 23
2.4.5 Sensibilidade 23
2.4.6 F1-Score 23
2.4.7 Área sob a curva ROC 24
2.4.8 Área sob a curva de precisão e *recall* 25
2.4.9 Validação Cruzada 26
2.4.10 Shapley Additive Explanation 26
3 TRABALHOS RELACIONADOS 28
3.1 Mortalidade Neonatal 28
3.2 Tratamento de Desbalanceamento de Classes 30
3.3 Diferencial da Pesquisa 31
4 METODOLOGIA 33
4.1 Coleta e Rotulação dos Dados 33
4.1.1 Descrição das Variáveis do Conjunto de Dados 35
4.2 Pré-processamento dos dados 36
4.3 Treinamento de Modelos com Aprendizado Supervisionado 38
5 RESULTADOS 41
6 CONCLUSÃO 46
REFERÊNCIAS 48

1 INTRODUÇÃO

A taxa de mortalidade infantil (TMI) é um grande indicador de desenvolvimento social de um país. Quando essa taxa é elevada, pode indicar condições precárias de vida e saúde, e baixo nível de desenvolvimento socioeconômico. A redução da mortalidade infantil é uma das metas do milênio da Organização das Nações Unidas (ONU) do qual muitos países do mundo fazem parte, inclusive o Brasil (World Health Organization, 2022). A redução da TMI ainda é um desafio ao redor do mundo, e no cenário brasileiro, apesar do declínio desse tipo de mortalidade, ela ainda é preocupante para a saúde pública do país.

O óbito neonatal é o falecimento que ocorre antes dos 28 dias completos de vida, sendo esse tipo de óbito o principal componente de mortalidade infantil desde a década de 80, pois representa entre 60% e 70% da mortalidade infantil em todo o Brasil até os dias atuais. A taxa de mortalidade neonatal(TMN) em 2007 é 13,2/1000 nascidos vivos (NV), que é bem mais elevada quando comparada com outros países como Argentina (10/1000), Chile (5/1000), Canadá (3/1000), Cuba (4/1000) e França(2/1000). A TMI do Brasil apresentou declínio no período de 1990 a 2015, passando de 47,1 para 13,3/1000 NV. Em 2016, observou-se um aumento desta taxa, passando para 14,0. De 2017 a 2019, voltou ao patamar de 2015, de 13,3/1000 NV (Secretaria de Vigilância em Saúde, 2021).

No ano de 2020, no estado do Rio Grande do Sul, a meta acordada para a TMI foi de 9,75/1000 NV, e no período, a taxa alcançada foi de 8,62/1000 NV, ou seja, a meta estadual para aquele ano foi atingida. Porém, em torno de 77,49% do total desses óbitos foram óbitos neonatais. Desse modo, o Rio Grande do Sul já alcançou a meta prevista pelo ODS-3 (Objetivos de Desenvolvimento Sustentável - Objetivo 3), e atualmente o desafio é manter as taxas dentro das metas estabelecidas e alcançadas, bem como, atingir a meta do Plano Estadual de Saúde (PES) de 2023 de 9,6/1000 NV (Secretaria da Saúde do Estado do Rio Grande do Sul, 2022).

A identificação de possíveis casos de óbitos neonatais e quais fatores estão associados com um aumento do risco de óbito neonatal é de suma importância para auxiliar na redução da TMI. Dentro de uma abordagem de ciência de dados, esse tipo de problema pode ser modelado utilizando aprendizado de máquina supervisionado, porém apresenta o desafio de ocorrer raramente, ou seja, a chance de ocorrer um óbito neonatal é significativamente menor que a chance de não ocorrer. Classificar a ocorrência ou não de um óbito neonatal é um problema de classificação desbalanceado por natureza, e para trabalhar

com modelos preditivos que lidam com conjunto de dados desbalanceados, algumas estratégias precisam ser adotadas para mitigar o efeito deste desbalanceamento(SáNCHEZ-HERNáNDEZ et al., 2019).

Em vista disso, esse trabalho tem o objetivo de avaliar diferentes estratégias para enfrentar conjunto de dados com classes desbalanceadas, um problema recorrente na área da saúde. Para esse propósito, foi utilizado um conjunto de dados relacionados a casos de mortalidade neonatal no estado do Rio Grande do Sul. Os dados foram obtidos a partir de bases que compõem o DATASUS (Departamento de Informática do Sistema Único de Saúde do Brasil), incluindo os dados de nascimentos da base SINASC (Sistema de Informações sobre Nascidos Vivos) e os dados de mortalidade da base SIM (Sistema de Informações sobre Mortalidade) (BELUZO et al., 2020).

Em relação às estratégias de modelagem de aprendizado de máquina, foram combinados quatro algoritmos de classificação e cinco estratégias para lidar com desbalanceamento de classes. Árvore de decisão, floresta aleatória, AdaBoost (do inglês, *Adaptive Boosting*) e XGboost (do inglês, *Extreme Gradient Boosting Trees*) foram os algoritmos utilizados para treinamento dos classificadores, e abordagem sensível ao custo, subamostragem aleatória, SMOTE (do inglês, *Synthetic Minority Oversampling Technique*), SMOTE-ENN (do inglês, *SMOTE Edited Nearest Neighbour*) e SMOTE-Tomek foram as estratégias escolhidas para lidar com o desbalanceamento de classes (DURAHIM, 2016). A eficácia das diferentes abordagens foi analisada através de diferentes métricas, mas principalmente comparando-se os valores das métricas de sensibilidade (*recall*), *f1-score* e área sob a curva de precisão e *recall*, pois essas métricas são sensíveis à ocorrência de falsos negativos no modelo. No problema em questão, é de suma importância que óbitos neonatais reais sejam identificados corretamente como tal, com a maior taxa possível de acerto para a classe positiva a fim de evitar subestimar o risco de óbito neonatal para novos casos.

O presente trabalho está estruturado em 6 capítulos. No Capítulo 2, será apresentada a fundamentação teórica necessária para que a metodologia e os resultados obtidos sejam compreendidos. No Capítulo 3, serão apresentados os trabalhos relacionados a este estudo, discutindo-se suas semelhanças e diferenças. No Capítulo 4, serão apresentadas as três principais etapas da metodologia adotada para os experimentos envolvidos no presente trabalho. No capítulo 5, serão apresentados os resultados e a discussão sobre esses resultados. Por fim, no Capítulo 6, serão apresentados as conclusões sobre os experimentos propostos e resultados obtidos neste trabalho.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo serão abordados os principais conceitos relacionados ao desenvolvimento de modelos de predição de óbito neonatal. Nesta perspectiva, este capítulo discorre sobre óbito neonatal, algoritmos de aprendizado de máquina para problemas de classificação, métodos para trabalhar com classes desbalanceadas, assim como os principais métodos de avaliação de modelos preditivos.

2.1 Óbito Neonatal

O óbito neonatal (ON) é o falecimento que ocorre nos primeiros 28 dias de vida (Jhpiego Corporation World Health Organization, 2016). O primeiro mês de vida é o período mais vulnerável para os recém nascidos, aproximadamente 75% dos ON ocorrem na primeira semana de vida, e em 2019 em torno de um milhão de recém nascidos faleceram nas primeiras 24 horas de vida (World Health Organization, 2022). A TMI é uma preocupação mundial há mais de três décadas (United Nations Children's Fund - UNICEF, 2015) e os óbitos neonatais, no Brasil, abrangem cerca de 70% dessa taxa (FRANçA; LANSKY, 2008), especificamente no estado do Rio Grande do Sul essa taxa foi em torno de 77.49% em 2020 (Secretaria da Saúde do Estado do Rio Grande do Sul, 2022).

2.2 Métodos de Aprendizado de Máquina

A inteligência artificial (IA) é um ramo da Ciência da Computação que simula a inteligência humana através de máquinas. Efetivamente, essa área busca reproduzir a inteligência humana de maneira mais eficiente. O aprendizado de máquina (AM) é uma área da IA que especificamente visa permitir que sistemas que aprendam autonomamente, sem intervenção humana (DORADO-DíAZ et al., 2019).

A AM é geralmente subdividido em três categorias: 1) aprendizado supervisionado; 2) aprendizado não-supervisionado; e 3) aprendizado por reforço. O aprendizado supervisionado é um método de aprendizado que consiste em receber dados rotulados e retornar uma predição com base nesses dados etiquetados, esse método possui dois tipos: regressão que prevê uma saída de valor contínuo; e classificação que categoriza dados em grupos discretos (GREENER et al., 2022). Já o aprendizado não-supervisionado com-

preende em receber dados não rotulados e produzir agrupamentos como saída. Por fim, o aprendizado por reforço tem como objetivo construir um sistema de recompensas para um agente a fim de permitir que o agente aprenda automaticamente ações que sejam mais adequadas em cada contexto (KOTSIANTIS; ZAHARAKIS; PINTELAS, 2007).

Neste trabalho, o interesse será em algoritmos de aprendizado supervisionado para classificação, uma vez que o objetivo é predizer quais casos enfrentam alto risco de óbito neonatal. Portanto, a seguir será discorrido sobre alguns algoritmos de aprendizado supervisionado para classificação que fazem parte do estado da arte para resolver esse tipo de problema, e por fim será discutido métodos de avaliação de modelos de AM.

2.2.1 Árvores de Decisão

As árvores de decisão são modelos de aprendizado de máquina supervisionados que utilizam um algoritmo que divide os dados de entrada com bases em testes e comparações de seus atributos. Essas divisões são realizadas com base um de dois critérios usuais: impureza de Gini e ganho de informação. Esses critérios buscam minimizar o número de testes até que uma classe seja determinada. O algoritmo de treinamento, a cada momento, determina qual o atributo que auxilia a forma máxima a reduzir a incerteza do modelo acerca da classe da instância. Nessas estruturas de árvore, as folhas representam rótulos de classe e os ramos representam a combinação de atributos que levam a esses rótulos de classe (DU et al., 2002). A impureza de Gini (entropia) de um conjunto de dados é um número que indica a probabilidade de novos dados aleatórios serem classificados erroneamente se receberem um rótulo de classe aleatório de acordo com a distribuição de classe no conjunto de dados. Um atributo com a menor impureza de Gini é selecionado para dividir o nó. Dado um conjunto de dados com "n" classes e a proporção de itens classificados como "p_i" no conjunto de dados, a Impureza de Gini é matematicamente definida da seguinte forma:

$$gini(p) = 1 - \sum_{i=1}^{n} p_i^2 \tag{2.1}$$

O ganho de informação mede a diferença na impureza dos dados antes e depois da divisão de um nodo da árvore, ou seja, dado um atributo x do nodo atual da árvore e S um conjunto de atributos dos possíveis filhos de x. O ganho de informação pode ser calculado da seguinte maneira:

$$g(S, x) = gini(x) - \sum_{i=1}^{S} gini(s_i) \tag{2.2}$$

As árvores de decisão estão entre os algoritmos de aprendizado de máquina mais populares devido à sua inteligibilidade e simplicidade. Um problemadas árvores de decisão é que elas potencialmente causam *overfitting* (em português, sobreajuste) dos dados, ou seja, a estrutura da árvore é muito dependente dos dados de treinamento e não representa com precisão a aparência dos dados no mundo real.

2.2.2 Florestas Aleatórias

Uma floresta aleatória (FA) é uma técnica de aprendizado de máquina, onde uma floresta contém muitas árvores de decisão que trabalham juntas para classificar novos pontos. Quando uma floresta aleatória é solicitada a classificar um novo ponto, ela fornece esse ponto a cada uma de suas árvores de decisão. Cada uma dessas árvores relata sua classificação e a floresta aleatória retorna a classificação mais popular. Algumas das árvores na floresta aleatória podem estar sobreajustadas (*overfitting*), mas ao fazer a previsão com base em um grande número de árvores, o sobreajuste terá menos impacto. As árvores em um classificador de floresta aleatória são criadas usando um subconjunto aleatório do conjunto de dados original com reposição, o que significa que um único ponto de dados pode ser escolhido mais de uma vez. Esse processo é conhecido como *bagging* e é util para minimizar o *overfitting*, dado que cada árvore individual é treinada em um subconjunto de dados originais (BREIMAN, 2001).

2.2.3 Adaptive Boosting

O *Adaptive Boosting*, popularmente conhecido como *AdaBoost*, é um dos métodos mais populares de *boosting*. Nesse método, o algoritmo adapta-se e tenta autocorrigir-se a cada iteração do processo. Inicialmente, todos os dados têm o mesmo peso, onde os pesos são alterados a cada resultado de uma árvore de decisão, dados classificados incorretamente recebem pesos maiores para serem corridos posteriormente, em uma próxima rodada, as rodadas ocorrem até que o limite de erro residual seja alcançado (BREIMAN, 2001). No geral, o *AdaBoost* é um tipo adequado de *boosting* para problemas de classificação.

2.2.4 XGBoost

O *XGBoost* (*Extreme Gradient Boosting Trees*), é um algoritmo de aprendizado de máquina com base em árvores de decisões e gradiente descendente. Ou seja, o algoritmo é composto por várias árvores de decisão, onde cada uma aplica o método de gradiente descendente que aprende com as outras árvores e, por fim, a classificação é obtida considerando as saídas de todas as árvores. Os grandes diferenciais do *XGBoost* são o bom gerenciamento de dados esparsos e sua flexibilidade para executar paralelamente ou distribuidamente (CHANG; CHANG; WU, 2018).

2.3 Desbalanceamento de Classes

O desbalanceamento de classes podem ser definido pela pequena incidência de uma categoria específica no atributo alvo de um conjunto de dados (classe minoritária) em comparação com as demais categorias (classes majoritárias). Os métodos para trabalhar com dados desbalanceados podem ser categorizados da seguinte forma: a) métodos de amostragem; b) modificações de algoritmos; e c) abordagem sensível ao custo. A seguir, essas abordagens serão detalhadas.

2.3.1 Métodos de Amostragem

O objetivo dos métodos de amostragem é balancear as classes dos conjuntos. Existem três técnicas de amostragem: 1) sobreamostragem (em inglês, *oversampling*); 2) subamostragem (em inglês, *undersampling*); e 3) combinação de subamostragem e sobreamostragem. Sobreamostragem envolve alterar o número total de itens de dados de classe aumentando a classe minoritária, enquanto a subamostragem reduz a classe majoritária (EBENUWA et al., 2019).

2.3.1.1 Subamostragem Aleatória

A subamostragem aleatória é uma técnica que elimina aleatoriamente os pontos amostrais da classe majoritária até que seu tamanho corresponda ao da classe minoritária. Nesse método também é possível obter uma distribuição de classe diferente de 50%, assim como é permitido subamostrar a classe minoritária para obter uma distribuição de classes

inferior à original (ALBISUA et al., 2013).

2.3.1.2 SMOTE

A SMOTE é uma técnica de sobreamostragem de minoria sintética (do inglês, *synthetic minority oversampling technique*) que gera novos pontos da classe minoritária com base nos "k" vizinhos mais próximos (ALBISUA et al., 2013). O processo inclui os seguintes passos:

1. escolha de dados aleatórios da classe minoritária;
2. cálculo da distância entre os dados aleatórios e seus k vizinhos mais próximos;
3. multiplicação da distância por um número aleatório entre 0 e 1 e adição do resultado à classe minoritária como uma amostra sintética;
4. iteração das etapas 2 e 3 até que a proporção desejada de classe minoritária seja atendida.

2.3.1.3 SMOTE-ENN

A SMOTE-ENN é uma técnica híbrida que combina o método SMOTE e o algoritmo de limpeza ENN (do inglês, *edited nearest neighbour*). A principal ideia desse método é reduzir o risco de *overfitting* no classificador devido à introdução de exemplos artificiais na classe minoritária, por isso, foi um dos métodos propostos para alcançar melhores resultados em conjuntos de dados com poucos exemplos minoritários (ALBISUA et al., 2013) (BATISTA; PRATI; MONARD, 2004). O método pode ser sintetizado a partir dos seguintes passos:

1. escolha de dados aleatórios da classe minoritária;
2. cálculo da distância entre os dados aleatórios e seus k vizinhos mais próximos;
3. multiplicação da distância por um número aleatório entre 0 e 1 e adição do resultado à classe minoritária como uma amostra sintética;
4. iteração das etapas 2 e 3 até que a proporção desejada de classe minoritária seja atendida;
5. determina k, como o número de vizinhos mais próximos;
6. encontra os k vizinhos mais próximos da observação entre as outras observações no conjunto de dados e retorne a classe majoritária do k vizinho mais próximo;
7. se a classe da observação e a classe majoritária do k vizinho mais próximo da ob-

servação forem diferentes, então a observação e seu k vizinho mais próximo serão excluídos do conjunto de dados;

8. iteração das etapas 2 e 3 até que a proporção desejada de cada classe seja cumprida.

2.3.1.4 SMOTE-Tomek

O SMOTE-Tomek é um método que combina SMOTE e o algoritmo de limpeza Links de Tomek, ou seja, o algoritmo gera dados sintéticos da classe minoritária usando o método SMOTE e remove os dados da classe majoritária que estão mais próximos da classe minoritária (BATISTA; BAZZAN; MONARD, 2003). O processo inclui os seguintes passos:

1. escolha de dados aleatórios da classe minoritária;
2. cálculo da distância entre os dados aleatórios e seus k vizinhos mais próximos;
3. multiplicação da distância por um número aleatório entre 0 e 1 e adição do resultado à classe minoritária como uma amostra sintética;
4. iteração das etapas 2 e 3 até que a proporção desejada de classe minoritária seja atendida;
5. escolha de dados aleatórios da classe majoritária;
6. se o vizinho mais próximo dos dados aleatórios forem os dados da classe minoritária (ou seja, link Tomek), remova o link Tomek.

2.3.2 Modificações de Algoritmos

A seguir, dois algoritmos com modificações para lidar com desbalanceamento de classes serão detalhados.

2.3.2.1 Floresta Aleatória Balanceada

A Floresta Aleatória Balanceada é um método *ensemble* no qual cada árvore da floresta receberá uma amostra *bootstrap* balanceada, ou seja, a cada iteração do algoritmo de floresta aleatória, uma amostra com a mesma proporção de classe é fornecida para a construção das árvores de decisões (CHEN; LIAW; BRIEMAN, 2004).

2.3.2.2 RUSBoost

O algoritmo *RUSBoost* que significa *Random Under Sampling Boosting* (em português, reforço de subamostragem aleatória). *RUSBoost* é uma combinação dos algoritmos subamostragem aleatória e *AdaBoost*. Durante o aprendizado, o problema de balanceamento de classe é amenizado pela subamostragem aleatória em cada iteração do algoritmo de reforço (SEIFFERT et al., 2010).

2.3.3 Abordagem Sensível ao Custo com Floresta Aleatória Ponderada

Na abordagem sensível ao custo, ao ajustar um modelo ao conjunto de dados de treinamento, busca-se minimizar o erro, considerando os custos da classificação correta ou incorreta, assim, o problema pode ser otimizado para minimizar o custo total da classificação incorreta. Nessa abordagem, a precisão não é tão importante quanto a implicação de classificar erroneamente um ponto (EBENUWA et al., 2019).

A Floresta Aleatória Ponderada é um método que segue a ideia de aprendizagem sensível ao custo. Para tanto, é atribuído um peso a cada classe, e a classe minoritária recebe o maior peso, ou seja, maior custo de erro de classificação. Os pesos das classes são incorporados no algoritmo de FA em dois momentos, para ponderar o critério de seleção de nós (por exemplo, o índice Gini), mas também nas folhas de cada árvore, ou seja, a predição de classe de cada folha é determinada por "voto majoritário ponderado". Assim, o voto ponderado de uma classe é o peso dessa classe vezes o número de casos dessa classe na folha. A previsão de classe final para uma FA é então determinada agregando o voto ponderado de cada árvore individual (CHEN; LIAW; BRIEMAN, 2004).

2.4 Avaliação de Modelos de Aprendizado de Máquina

No desenvolvimento de um modelo de aprendizado de máquina, após o treinamento do modelo em questão é necessário avaliar seu poder preditivo, para isso, existem diferentes métricas de desempenho. A avaliação do modelo pode ser considerada a garantia de qualidade do aprendizado de máquina. A avaliação adequada do desempenho do modelo em relação às métricas e requisitos determina como o modelo funcionará no mundo real. Na sequência são descritas algumas técnicas de avaliação de modelos de AM amplamente usadas em problemas de classificação e adotadas nestes trabalho.

2.4.1 Matriz de Confusão

A matriz de confusão é uma das ferramentas mais intuitivas para analisar a corretude e a precisão do modelo. Essa matriz é utilizada para observar os acertos e erros do modelo em relação aos valores reais. Para classificações com n classes, a matriz é do tipo nxn, onde a diagonal principal dessa matriz é composta pelas classificações corretas. As classificações corretas são dadas pelas métricas de Verdadeiros Positivos e Verdadeiros Negativos, já as classificações com erro são do tipo Falso Positivo e Falso Negativo (MONARD; BARANAUSKAS, 2003). Os tipos de acertos e erros de classificação são detalhados a seguir:

- Verdadeiros Positivos (TP do inglês *True Positives*): ocorrem quando a classificação prevê que um valor pertence a classe positiva e o valor realmente pertence a classe positiva.
- Verdadeiros Negativos (TN do inglês *True Negatives*): ocorrem quando a classificação prevê que um valor pertence a classe negativa e o valor realmente pertence a classe negativa.
- Falsos Positivos (FP do inglês *False Positives*): ocorrem quando a classificação prevê que um valor pertence a classe positiva, porém o valor pertence a classe negativa.
- Falsos Negativos (FN do inglês *False Negatives*): ocorrem quando a classificação prevê que um valor pertence a classe negativa, porém o valor pertence a classe positiva.

A Figura 2.1 mostra um exemplo genérico de uma matriz de confusão para um problema de classificação binário, similar ao abordado neste trabalho. Sempre haverá algum erro associado a cada modelo criado, então não existe uma regra rígida que diga qual tipo de erro deve ser minimizado em todas as situações. Isso dependerá puramente das necessidades do negócio e do contexto do problema a ser resolvido. No caso deste trabalho, objetiva-se capturar o máximo possível de casos de óbitos neonatais, já que uma predição de um caso que não acarretará em óbito, não é tão perigosa quanto não prever corretamente um óbito ocorrido.

Figura 2.1 – Matriz de Confusão para um problema binário

Classe	predita C_+	predita C_-
verdadeira C_+	Verdadeiros positivos T_P	Falsos negativos F_N
verdadeira C_-	Falsos positivos F_P	Verdadeiros negativos T_N

Fonte: (MONARD; BARANAUSKAS, 2003)

2.4.2 Acurácia

A acurácia é uma métrica que indica a taxa de predições corretas realizadas pelo modelo, como segue a fórmula abaixo:

$$acuracia = \frac{TP + TN}{TP + FP + TN + FN} \tag{2.3}$$

Essa métrica é útil quando o problema tem classes balanceadas, pois em caso de classes desbalanceadas, essa métrica pode resultar em falsa impressão de que o modelo está atingindo um bom desempenho preditivo (FERNáNDEZ et al., 2018)).

2.4.3 Acurácia Balanceada

Em problemas com classes desbalanceadas, a acurácia não é uma boa métrica a ser usada, pois pode mascarar baixos resultados para verdadeiros positivos. A acurácia balanceada é uma alternativa para casos de desbalanceamento de dados, porque não é influenciada pelo desbalanceamento das classes, posto que os cálculos ocorrem em cima da taxa de verdadeiros positivos e da taxa de verdadeiros negativos (TRAN; LE; SHI, 2022), como demonstrado na fórmula a seguir:

$$acuracia\_balanceada = \frac{1}{2}\left(\frac{TP}{TP + FN} + \frac{TN}{TN + FP}\right) \tag{2.4}$$

2.4.4 Precisão

A precisão é uma métrica que indica a probabilidade do modelo ter classificado corretamente uma instância da classe positiva, ou seja, é a proporção de instâncias que realmente pertencem à classe positiva considerando todas que foram classificadas como tal pelo modelo, conforme definido na fórmula abaixo:

$$precisao = \frac{TP}{TP + FP} \tag{2.5}$$

Se o objetivo do problema for minimizar os falsos positivos, então essa métrica deve ser o mais próximo possível de 100% (FERNáNDEZ et al., 2018).

2.4.5 Sensibilidade

A métrica sensibilidade, também conhecida como *recall*, indica a proporção de casos positivos que foram identificados corretamente pelo modelo. É também conhecida como taxa de verdadeiros positivos do modelo e pode ser calculada conforme a fórmula abaixo:

$$sensibilidade = \frac{TP}{TP + FN} \tag{2.6}$$

Se o foco do problema é minimizar os falsos negativos, então essa métrica deve ser o mais próximo possível de 100% já que essa métrica retorna a proporção de acertos em relação à classe positiva (FERNáNDEZ et al., 2018).

2.4.6 F1-Score

A métrica *F1-Score* representa um compromisso entre as métricas de precisão e sensibilidade, sendo calculada uma média harmônica entre ambas. Essa é uma boa métrica em casos de classes desbalanceadas e quanto maior a taxa resultante dessa métrica melhor é o modelo(FERNáNDEZ et al., 2018). A fórmula a seguir apresenta o cálculo de F1-Score:

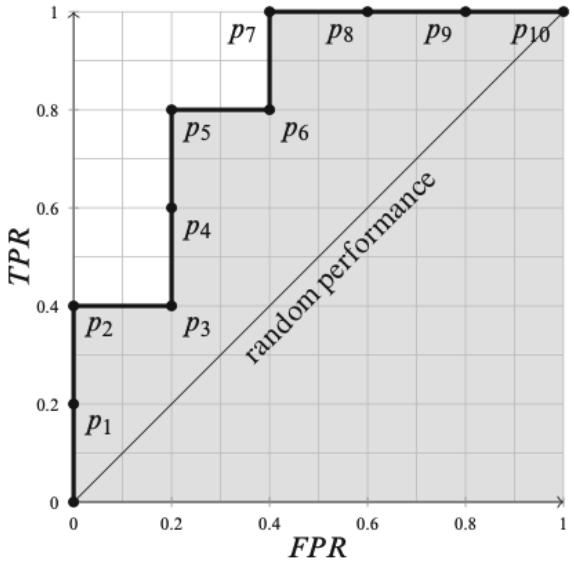
$$F1 = 2 * \frac{precisao * sensibilidade}{precisao + sensibilidade} \tag{2.7}$$

2.4.7 Área sob a curva ROC

A métrica de área sob a curva (AUC, em inglês *Area Under the Curve*) calculada a partir da curva ROC (em inglês *Receiver Operating Characteristics*) é usualmente adotada para a avaliação de um modelo de aprendizado de máquina. A curva ROC indica a eficiência do modelo criado em distinguir entre duas coisas que podem ser 1 ou 0, ou positivo e negativo. Essa curva utiliza dois parâmetros: a taxa de verdadeiro positivos (TPR do inglês, *true positive rate*), que é dada por $TP/(TP + FN)$ e a taxa de falso positivos (FPR do inglês, *false positive rate*), que é dada por $FP/(FP + TN)$. Portanto, uma curva ROC traça "Taxa de Verdadeiro Positivos vs. Taxa de Falso Positivos", em diferentes limiares de classificação (FERNáNDEZ et al., 2018). A AUCROC visa simplificar a análise de ROC, ou seja, a AUCROC resume a curva ROC em um único valor, agregando todos os limiares da curva ROC. O valor do AUCROC varia de 0 até 1, sendo o limiar entre as classes igual ao valor de 0,5, e quanto maior a AUCROC melhor o modelo.

A Figura 2.2 mostra um exemplo genérico de uma curva ROC.

Figura 2.2 – Exemplo de uma curva ROC



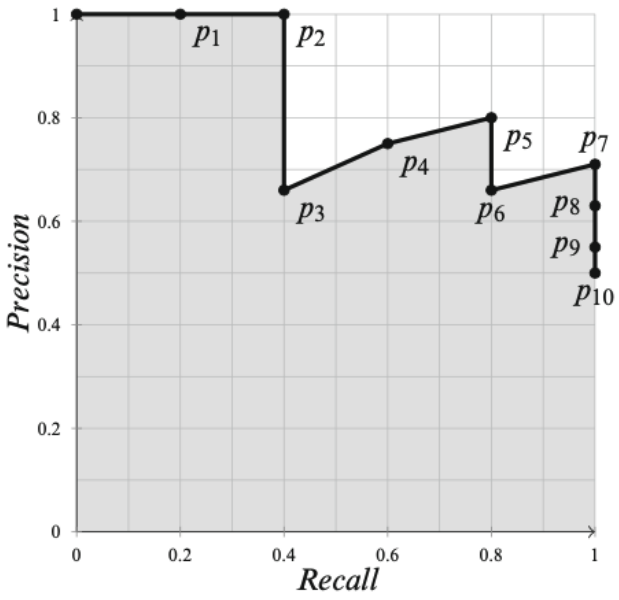
Fonte: (FERNáNDEZ et al., 2018)

2.4.8 Área sob a curva de precisão e *recall*

Essa métrica é calculada de forma análoga à área sob a curva ROC, mas é considerada mais adequada para problemas com classes desbalanceadas. Como citado anteriormente, existe uma compensação na busca por melhorar a precisão e a sensibilidade (*recall*) de um modelo. Consequentemente, essa compensação também pode ser representada graficamente gerando uma curva e calculando a área sob esta curva, denotada por AUPRC (do inglês, *Area Under the Precision-Recall Curve*). Um valor alto de AUPRC representa alta sensibilidade e alta precisão, onde alta precisão está relacionada a uma baixa taxa de falsos positivos e alta sensibilidade está relacionada a uma baixa taxa de falsos negativos. O limiar da curva precisão-recall é determinado pela proporção de positivos(P) e negativos(N) como $y = P/(P + N)$. Devido a essa linha de base móvel, a AUPRC também muda com a relação $P : N$ (FERNáNDEZ et al., 2018).

A Figura 2.3 mostra um exemplo de uma curva PR.

Figura 2.3 – Exemplo de uma curva precisão-recall



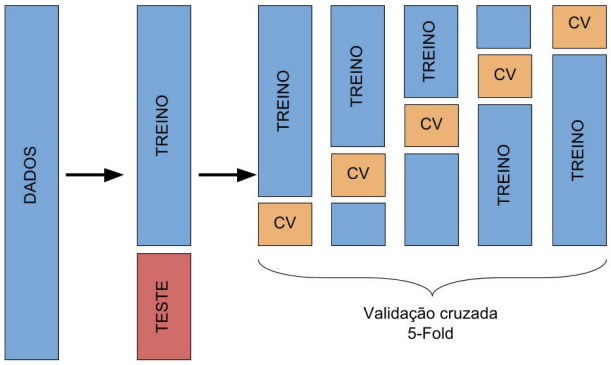
Fonte: (FERNáNDEZ et al., 2018)

2.4.9 Validação Cruzada

A validação cruzada *k-fold* é uma técnica utilizada para detectar sobreajuste em modelos. Essa técnica avalia modelos de AM através de vários modelos que utilizam subconjuntos dos dados de entrada disponíveis, ou seja, os dados de entrada são divididos em k subconjuntos de dados disjuntos, onde um modelo é treinado em todos, menos em um (k-1) dos conjuntos de dados e, em seguida, o modelo é avaliado no conjunto de dados que não foi usado para treinamento. O viés do método *k-fold* diminui a medida que k aumenta, porém, um valor de k elevado aumenta o custo computacional, além de implicar no aumento de variância. Os valores mais usuais para k são 2, 5 ou 10 (EFRON; TIBSHIRANI, 1983).

A Figura 2.4 apresenta um exemplo do processo de *k-fold* com $k = 5$.

Figura 2.4 – validação cruzada 5-fold



Fonte: Laboratório de Estatística e Geoinformação (2023)

2.4.10 Shapley Additive Explanation

O *Shapley Additive Explanations* (SHAP) é uma abordagem da teoria dos jogos que é utilizada em AM para auxiliar na explicação da saída de um modelo. O valor de Shapley é um método que calcula a contribuição de cada membro dentro de um grupo para alcançar determinado resultado, ou seja, dado um grupo de agentes que trabalham em conjunto para realizar uma tarefa que tenha uma recompensa diretamente proporcional ao esforço, busca-se responder o quanto cada agente contribuiu para alcançar o objetivo. Sendo assim, esse valor é calculado através da soma ponderada da diferença entre os mo-

delos em que um atributo foi incluído e nos modelos no qual não foi(LUNDBERG et al., 2017). Finalmente, com os valores de Shapley calculados para cada atributo do modelo é possível entender o quanto cada atributo tem importância no modelo em questão.

3 TRABALHOS RELACIONADOS

Diversas pesquisas demonstraram o grande valor que o aprendizado de máquina pode agregar na predição de possíveis óbitos neonatais, assim como os principais fatores que podem ter um grande impacto nesse desfecho. Neste capítulo serão apresentadas pesquisas relacionadas à aplicação de aprendizado de máquina para predizer ocorrências de óbitos neonatais e também pesquisas que discorrem sobre métodos para trabalhar com problemas que possuem classes desbalanceadas.

3.1 Mortalidade Neonatal

No artigo *"Assessing the performance of machine learning models to predict neonatal mortality risk in Brazil, 2000-2016"* (ALVES et al., 2020), os autores tiveram como finalidade investigar a associação entre características relacionadas e risco de mortalidade neonatal no Brasil. Para o estudo, foram coletados dos sistemas SIM e SINASC, registros de nascidos vivos e registros de óbitos neonatais, respectivamente, no período de 2006 a 2016 que ocorreram em todo o território brasileiro. Para relacionar os registros entre duas bases dados foi utilizado o número identificador de óbito, então após o pré-processamento dos dados, cinco algoritmos de aprendizados de máquina foram aplicados para obter os modelos de predição, os algoritmos utilizados foram florestas aleatórias (FA), XGBoost e máquina de vetores de suporte (SVM). Posteriormente ao treinamento e testes dos modelos, as métricas matriz de confusão, acurácia, sensibilidade, especificidade e área sob a curva característica de operação do receptor (AUROC) foram calculadas para verificar a qualidade dos modelos. Ao final, foi constatado que o modelo que utilizou XGBoost obteve a melhor performance com 93,93% de ROC AUC e os fatores que obtiveram maior relevância para detectar um óbito neonatal foram: peso do recém-nascido, escala de *apgar*(avaliação do recém-nascido à vida extrauterina) no primeiro e quinto minuto, malformações congênitas, idade gestacional e número de consultas de pré-natal.

O trabalho *"Machine learning to predict neonatal mortality using public health data from São Paulo - Brazil"* (BELUZO et al., 2020), propôs pesquisar e analisar os fatores específicos relacionados à mortalidade neonatal, usando técnicas de aprendizado de máquina em contrapartida aos estudos demográficos usuais realizados no Brasil. Nessa pesquisa, também foram coletados do SIM e do SINASC, registros de nascidos vivos e registros de óbitos neonatais, porém nesse estudo os registros são do período entre 2012 e

2018, porém apenas de nascimentos e óbitos neonatais ocorridos na cidade de São Paulo. Para relacionar os registros entre as duas bases de dados foi utilizado o número de declaração de nascido vivo (DNV), posteriormente os dados foram pré-processados e os treinamentos e testes dos modelos foram realizados. Os algoritmos de aprendizado de máquina utilizados neste estudo foram SVM, regressão logística (RL), FA e XGBoost, mas também as métricas matriz de confusão, ROC, AUROC e explicações aditivas Shapley (SHAP) foram utilizadas para avaliar a performance dos modelos de predição. Ao final, foi demonstrado que o algoritmo XGBoost obteve o melhor desempenho para classificar óbitos neonatais com 87,2% de AUROC, inclusive o método é capaz de fornecer tanto uma resposta de risco de morte quanto uma interpretação do resultado obtido e os resultados apontam peso do recém-nascido, malformação congênita, escala de *apgar* no primeiro e quinto minuto e tipo de gestação como fatores mais significativos para óbito neonatal.

Na pesquisa *"Neonatal mortality prediction with routinely collected data: a machine learning approach"* (BATISTA et al., 2021), os autores sugeriram predizer o risco de mortalidade neonatal utilizando apenas dados disponíveis do cotidiano na cidade de São Paulo. Os dados também foram coletados do SIM e do SINASC, ou seja, registros de todos os nascidos vivos de óbitos neonatais ocorridos no município de São Paulo entre 2012 e 2017 foram utilizados neste estudo. Um relacionamento probabilístico foi realizado para conectar os registros das duas bases e, especificamente nesta pesquisa, todos os registros de nascimentos ocorridos entre 2012 e 2016 foram usados para treinar os algoritmos de aprendizado de máquina, enquanto os registros de nascimentos ocorridos em 2017 foram utilizados para testar a desempenho preditivo dos mesmos. Para os modelos preditivos foram utilizados os seguintes algoritmos: RL, redes neurais (RN) e XGBoost. Diversas métricas foram utilizadas para avaliá-los, como f1-score, valor preditivo negativo, área sob a curva de sensibilidade e precisão (AUPRC), SHAP, também foi utilizada a porcentagem do total de mortes incluída entre os 5% de risco previsto mais alto e o risco previsto de 5% mais baixo e o algoritmo com melhor desempenho foi então aplicado separadamente para subgrupos vulneráveis. O diferencial desse artigo foi a diversidade de métricas utilizadas para a validação dos modelos. Como resultados, os cinco fatores que demonstraram mais importância na predição foram escala de *apgar* de primeiro minuto e quinto minuto, peso ao nascer, presença de anomalia congênita e idade gestacional, o melhor desempenho foi obtido pelo algoritmo XGBoost com 97% de AUROC e 55% de f1-score, e não houveram diferenças significativas no desempenho preditivo para subgru-

pos vulneráveis.

3.2 Tratamento de Desbalanceamento de Classes

No artigo *"Predictive Modeling of ICU Healthcare-Associated Infections from Imbalanced Data. Using Ensembles and a Clustering-Based Undersampling Approach"* (SáNCHEZ-HERNáNDEZ et al., 2019), os pesquisadores propuseram auxílio na tomada de decisões para reduzir a taxa de infecções. Nessa área é necessário construir classificadores confiáveis a partir de conjuntos de dados desbalanceados. No trabalho foi proposto uma estratégia de subamostragem baseada em agrupamento para ser usada em combinação com classificadores ensemble, portanto um estudo comparativo foi realizado para validar a proposta. Dos dados de 4.616 pacientes da UTI do Hospital Universitário de Salamanca, 6,7% do total com infecções contraídas, foram submetidos a vários classificadores simples e ensemble tanto ao conjunto de dados original quanto aos dados pré-processados por meio de diferentes métodos de reamostragem. Como classificadores simples, foram testados árvores de decisão (AD), redes bayesianas (RB) e SVM, já os *ensemble* aplicados foram árvore aleatória, FA, ensacamento (em inglês, *bagging*), estímulo adaptativo (AdaBoost do inglês *Adaptive Boosting*) e *random committee*. Como métodos de amostragem foram utilizados SMOTE (técnica de sobreamostragem minoritária sintética), subamostragem aleatória(RUS), subamostragem por clusterização. Os resultados foram analisados por meio de métricas clássicas e recentes projetadas especificamente para classificação de dados desbalanceados, como precisão, sensibilidade, F-score, AUCROC, média geométrica e precisão otimizada. O estudo evidenciou que o método de subamostragem por clusterização obteve melhores resultados em comparação com RUS e SMOTE.

No trabalho *"An effective up-sampling approach for breast cancer prediction with imbalanced data: A machine learning model-based comparative analysis"* (TRAN; LE; SHI, 2022), foi proposto o método de *engineered up-sampling* (ENUS) para manipular as classes desbalanceadas presentes nos registros de câncer de mama com o objetivo de aprimorar o desempenho preditivo dos modelos de aprendizado de máquina. Nos resultados experimentais foram mostrados que quando a razão da classe minoritária com a classe majoritária é inferior a 20%, os modelos de treinamento com ENUS melhoraram a precisão balanceada em 3,74%, a sensibilidade em 8,36% e o score F1 em 3,83%. No estudo foi identificado que o XGBoost usando ENUS alcançou o melhor desempenho com uma

precisão média balanceada de 97,47%, sensibilidade de 97,88%, e score F1 de 96,20%. O conjunto de dados possuía um total de 682 dados, onde 64.96% pertenciam à classe negativa e 35.04% à classe positiva. Os classificadores utilizados foram: k-vizinhos mais próximos (knn), árvore de decisão (AD), FA, RN, redes neurais multicamadas (RNM),SVM e XBoost. Acurácia, acurácia balanceada, sensibilidade, especificidade, precisão e score F1 foram as métricas utilizadas para avaliar os modelos. Portanto, foi demonstrado que os modelos floresta aleatória e redes neurais tiveram o menor tempo de treino e o XGBoost apresentou melhor desempenho. O trabalho forneceu uma comparação abrangente de uma ampla gama de métodos de aprendizado de máquina na previsão do risco de câncer de mama e pode ser usado como uma ferramenta para profissionais de saúde para detectar de forma eficaz o câncer de mama.

Já no artigo *"Comparison of sampling techniques for imbalanced learning"* (DURAHIM, 2016), foram comparados algoritmos de amostragem em relação aos seus tempos de execução e precisões de classificação obtidos de diferentes classificadores. Para o estudo, foram utilizados dois conjuntos de dados disponibilizados pela Universidade da Califórnia, o primeiro conjunto de dados é relacionado à avaliação de carros, com 1728 dados e como taxa de desbalanceamento de 1/25, já o segundo conjunto de dados é sobre tomografias cardíacas, com 267 registros e com taxa de desbalanceamento de 1/4. Três algoritmos de classificação foram utilizados: SVM, FA e knn. Doze métodos de amostragem foram usados, sete de subamostragem e cinco de sobreamostragem. Foi demonstrado que as acurácias de classificações utilizando os métodos de sobreamostragem foram superiores aos métodos de subamostragem. Em relação aos tempos de execução, os métodos de amostragem foram semelhantes, no entanto, as classificações mais eficientes foram as acompanhadas de métodos de subamostragem. Entre os algoritmos de amostragem propostos, o método ADASYN (em português, amostragem sintética adaptável) foi a melhor escolha tanto nos tempos de execução, no aumento no tamanho dos dados e no desempenho das classificações.

3.3 Diferencial da Pesquisa

Conforme os estudos citados, existem muitas pesquisas que têm o objetivo de predizer, utilizando modelos de aprendizado de máquina supervisionados, quais fatores possuem uma maior impacto na mortalidade neonatal. No entanto, a presente pesquisa se difere dos demais trabalhos mencionados nos seguintes pontos:

- Analisar diferentes métodos para resolver problemas acentuados de desbalanceamentos de dados;
- Comparar algoritmos de classificação baseados em árvores para resolver problemas acentuados de desbalanceamentos de dados;
- Avaliar o desempenho de modelos preditivos em identificar o risco de óbito neonatal.

4 METODOLOGIA

A metodologia proposta neste trabalho segue os três principais passos descritos a seguir:

- Coleta e Rotulação dos Dados: consiste na etapa em que os dados são coletados das bases SINASC e SIM do DATASUS, posteriormente os registros entre as bases são linkados e variáveis desnecessárias ou inconsistentes são removidas do conjunto de dados;
- Pré-processamento dos Dados: é a etapa onde o conjunto de dados é preparado para posteriormente ser utilizado nas modelagens;
- Amostragem e Aprendizado Supervisionado: abrange a etapa de treinamento e validação dos modelos de aprendizado de máquina.

4.1 Coleta e Rotulação dos Dados

Essa etapa do experimento foi implementada utilizando a linguagem de programação R, juntamente com as bibliotecas reclin, microdatasus e dplyr. Para o trabalho, foram coletados inicialmente os registros de nascimentos e óbitos ocorridos no território brasileiro no período entre 2011 e 2020. Foi escolhido o período de dez anos, pela possível variabilidade dos valores dos registros, onde os anos de 2021 e 2022 foram excluídos por ocorrem no período pós COVID-19. Essas informações foram coletadas do DATASUS, que é o sistema de informática do Sistema Único de Saúde (SUS) responsável pela coleta, processamento e disseminação de informações sobre a saúde no Brasil. Os dados foram coletados de duas bases diferentes, as bases SINASC (Sistema de Informação sobre Nascidos Vivos) e SIM (Sistema de Informação sobre Mortalidade), utilizando o pacote em R chamado microdatasus, este pacote apresenta funções para download dos arquivos dos dados do DATASUS. Ao utilizar o método que processa os dados coletados do SIM dentro do período entre 2011 à 2013, a função retorna um erro ocasionado pela presença de três valores duplicados na tabela de códigos da cidade/país de naturalidade, então ao realizar o mapeamento do campo de naturalidade dos registros, os registros que contêm algum dos valores que tem código duplicado, o método encerra o processamento e retorna uma exceção, para contornar o problema foi necessário a remoção dos valores duplicados da tabela com problema antes de executar o método de processamento de dados do SIM. Foi

confirmado com responsável pela biblioteca que essa situação é um problema que deve ser resolvido futuramente e que a solução encontrada para evitar o problema é adequada.

É importante ressaltar que apenas dados públicos desses sistemas foram fornecidos, conforme as diretrizes da LGPD (Lei Geral de Proteção de Dados).

Após a coleta, para diminuir o volume de dados e a complexidade da associação entre os registros das duas bases, os registros de óbitos que ocorreram após 28 dias de vida foram eliminados. Também foram eliminados registros de óbitos cujas datas de nascimento eram anteriores ao ano de 2011.

Como os registros de nascimentos e de óbitos foram obtidos de dois sistemas diferentes, para o experimento é necessário que esses registros sejam unidos. Esta etapa é desafiadora, por não haver a disponibilidade de identificadores únicos representando cada indivíduo. Portanto, para realizar essa união, foi utilizado o pacote em R chamado Reclin que implementa metodologia para vincular registros com base em chaves inexatas (BELUZO et al., 2020). Para a união, foram fornecidos os campos "Data de nascimento", "Sexo", "Idade da mãe", "Código do município de residência", "Tipo da Gravidez", "Tipo de parto", "Raça/Etnia", que são os campos com as menores taxas de valores faltantes nos registros coletados, como pode ser visto na Tabela 4.1. Para determinar possíveis pares de registros que pertencem a mesma pessoa, foi utilizado um método que pontua os pares com base nos vetores de comparação e seleciona aqueles com uma pontuação acima de algum limite. Nesse trabalho, foi escolhido um limite de valor igual à quatro, que significa que no máximo um dos campos pode ser diferente entre os pares de registros. Para identificar os pares a serem unidos foi usado o método que seleciona pares a partir da pontuação mais alta, onde os pares são selecionados apenas quando cada um dos registros em um par não foi selecionado anteriormente.

Tabela 4.1 – Porcentagem de valores ausentes nos campos utilizados para realizar a união de registros entre as bases *SINASC* e *SIM*

Campo	**Porcentagem de Valores Ausentes**
Data de nascimento	0%
Sexo	12%
Idade da mãe	0%
Escolaridade da mãe	2%
Código do município de residência	0%
Tipo de gravidez	0%
Tipo de parto	0%
Raça/Etnia	5%

Visto que o tamanho dos conjuntos de dados a serem ligados era muito grande, foi decidido realizar o *linkage* por ano de nascimento, portanto, o processo de *linkage*

foi realizado em dez iterações, uma para cada ano desejado para o trabalho. Em média 38,27% dos registros de óbitos foram ligados a registros de nascimentos com sucesso, como mostra a Tabela 4.2. O *linkage* com os registros de 2014, não obteve mais do que 1% de sucesso, ou seja, 1% dos registros das bases tinham registros com valores em comum, foi investigado o porquê dessa diferença no resultado para os registros desse ano, especificamente, porém nenhuma solução foi encontrada.

Tabela 4.2 – Porcentagem de *linkages* realizados com sucesso entre as bases *SINASC* e *SIM*

Ano	**Porcentagem registros ligados**
2011	47,04%
2012	35,89%
2013	35,76%
2014	00,54%
2015	38,47%
2016	39,28%
2017	40,07%
2018	40,56%
2019	52,71%
2020	52,35%
Média de *linkages*	38,27%

Após o processo de união, para rotular as amostras como óbito neonatal ou não, foi adicionado um novo campo ao conjunto de dados resultante. Isso foi feito identificando os registros que possuíam os campos de data de nascimento e de data de óbito. Em seguida, todos os registros de óbitos que não foram unidos, foram removidos do conjunto de dados. Como os dados do SIM foram utilizados apenas para permitir a rotulação do conjunto de dados, os campos pertencentes ao SIM foram removidos. Também foram identificados a taxa de valores faltantes para cada campo do conjunto de dados resultantes, assim sendo, todos os campos com mais de 20% de valores faltantes foram removidos da base de dados, assim como outros campos considerados desnecessários para o experimento. Desta forma, o conjunto de dados resultante possui 27 atributos e 24.283.682 registros.

4.1.1 Descrição das Variáveis do Conjunto de Dados

Nesta seção será detalhada os atributos do conjunto de dados, assim como algumas informações estatísticas sobre esses dados.

Os atributos do conjunto de dados podem ser classificados nas seguintes categorias: 1) condições socioeconômicas maternas; 2) características obstétricas maternas; 3) características relacionadas ao recém-nascido; e 4) características pré-natais.

1. Condições socioeconômicas maternas: idade, estado civil, escolaridade e raça/etnia.
2. Características obstétricas maternas: número de nascidos vivos, número de óbitos fetais, número de gestações anteriores, número de cesáreas e partos normais e tipo de gravidez (simples, dupla, tripla, etc).
3. Características relacionadas ao recém-nascido: peso, raça/etnia, número de semanas de gestação, escala de Apgar no primeiro minuto, escala de Apgar no quinto minuto, tipo de apresentação (cefálico, pélvica, transversa) e indicador de anomalia congênita
4. Características pré-natais: número de consultas pré-natais, mês de gestação em que iniciou o pré-natal, número de consultas de pré-natal, local de ocorrência do nascimento, classificação do grupo de Robson, tipo de parto, informação se parto foi induzido e se cesárea ocorreu antes do parto iniciar.

Por fim, os dados foram separados em dois conjuntos, o primeiro conjunto de dados de testes que inclui registros de nascimentos entre o período de 2011 e 2018, e o conjunto de dados de testes que inclui registros dos anos de 2019 e 2020.

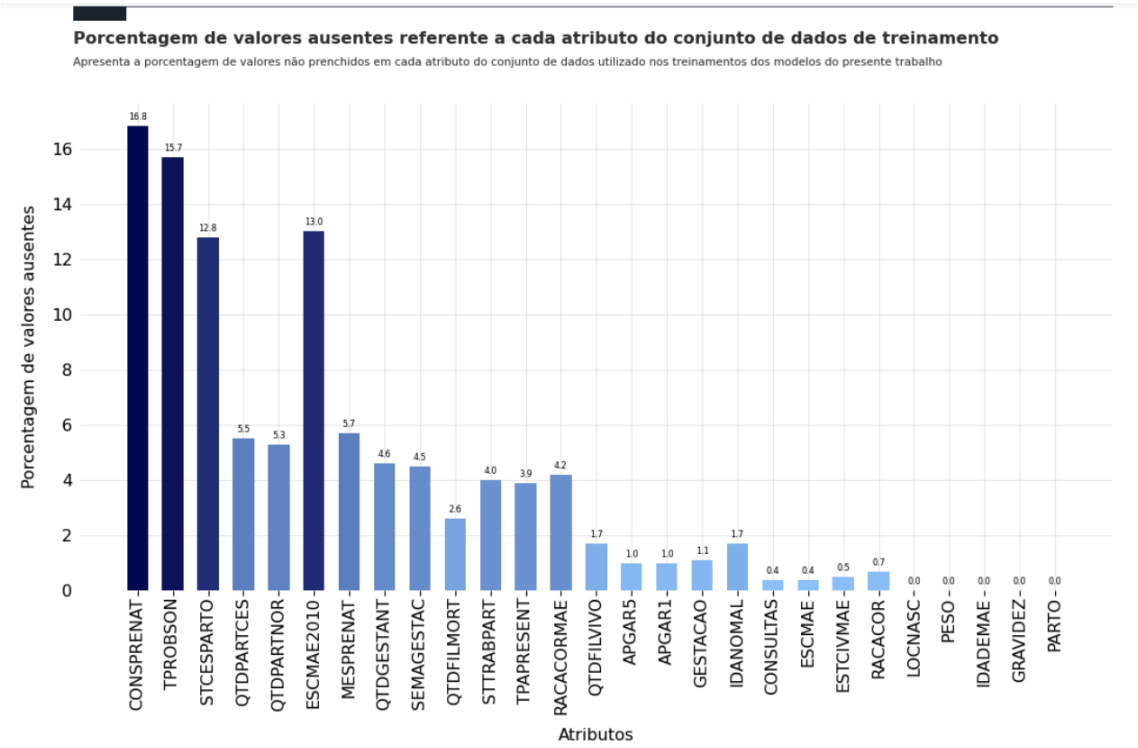
4.2 Pré-processamento dos dados

O pré-processamento de dados, antes de treinar um modelo de aprendizado de máquina, é uma etapa fundamental para conseguir extrair o melhor de cada modelo. No contexto dos dados de saúde pública brasileira, a ocorrência de dados ausentes ou inconsistentes é comum e ocorre principalmente devido ao preenchimento incorreto de formulários.

A seguir, as Figuras 4.1 e 4.2 apresentam com mais detalhes as taxa de valores ausentes nos atributos utilizados nos conjuntos de treinamento e de testes desse trabalho. A taxa de valores ausentes entre os conjuntos de dados de treino e teste variam, onde o conjunto de testes possui taxas de valores ausentes menores que o conjunto de treino na maioria dos campos, isso ocorre porque ao longo do tempo os campos são preenchidos com mais frequência.

Embora a remoção de grande parte dos atributos com muitos valores ausentes tenha sido realizada, alguns permaneceram. Porém, para serem usados nos algoritmos de aprendizado de máquina, precisamos que todos os valores sejam preenchidos. Portanto, foram aplicadas duas técnicas distintas para preencher valores faltantes, na primeira téc-

Figura 4.1 – Relação de valores ausentes para cada atributo do conjunto de dados de treinamento



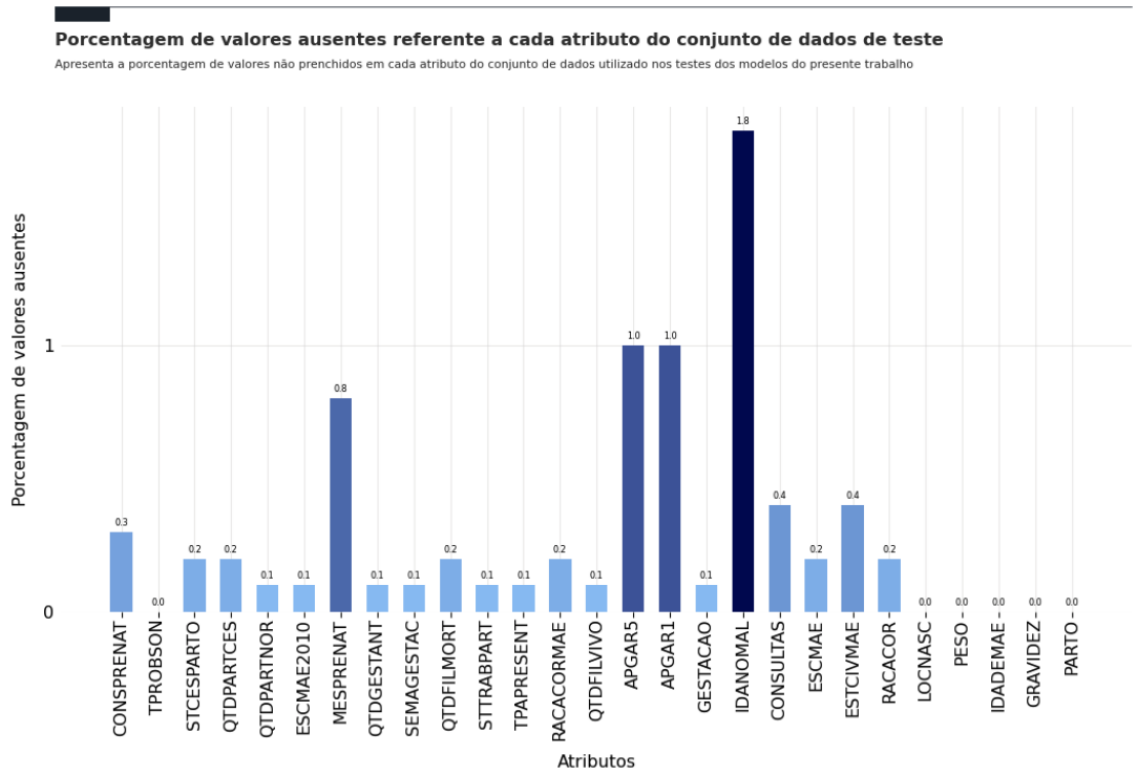
Fonte: A autora

nica os campos com valores numéricos contínuos foram preenchidos utilizando a mediana calculada através de todos os valores presentes para o campo em questão. Na segunda técnica os campos não numéricos foram preenchidas usando o valor mais frequente para presente no campo em foco(BELUZO et al., 2020).

Para atributos numéricos contínuos, é necessário posicioná-los em uma mesma escala para minimizar influências indevidas de um atributo sobre o outro. Para tanto, a técnica *MinMax Scaler* foi utilizada, na qual cada um dos atributos é ajustado para encaixar-se em um intervalo entre 0 e 1.

Os atributos categóricos não numéricos foram codificados para valores numéricos únicos através do método *Ordinal Encoder*. Ou seja, cada categoria de um determinado atributo foi mapeada para um valor inteiro específico. Esse mapeamento foi necessário porque alguns algoritmos de aprendizado de máquina não interpretam corretamente dados discretos.

Figura 4.2 – Relação de valores ausentes para cada atributo do conjunto de dados de teste



Fonte: A autora

4.3 Treinamento de Modelos com Aprendizado Supervisionado

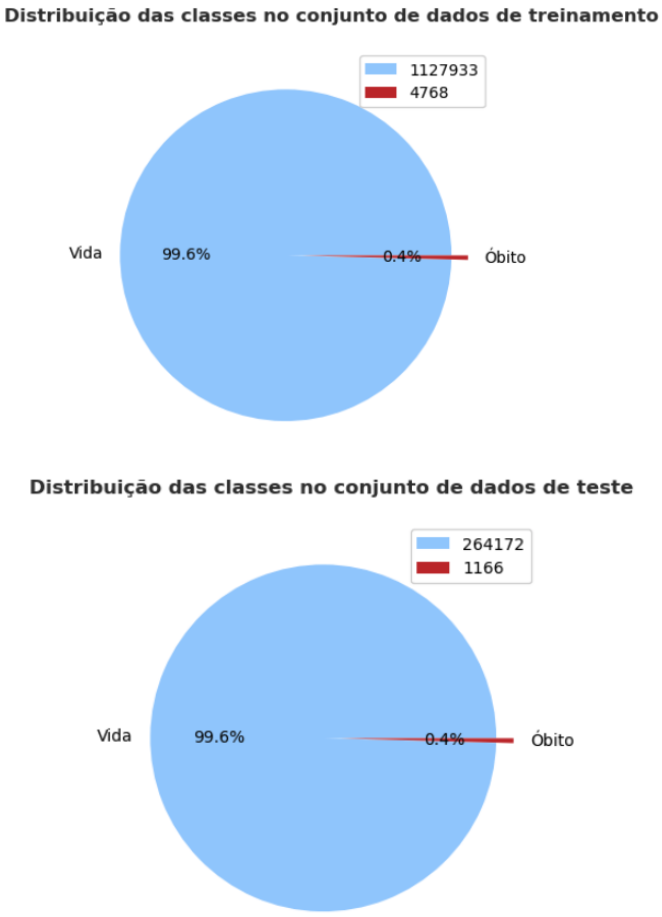
O treinamento e validação dos modelos nos experimentos realizados foram implementados utilizando a linguagem de programação *Python*, juntamente com as bibliotecas *Scikit-Learn*, *ImbLearn*, *XGBoost*, *Pandas* e *MatplotLib*.

O conjunto de dados original que inclui registros de nascimento em todo o território brasileiro possui 24.283.682 registros. Entretanto, por limitações de tempo de execução para realizar os experimentos, o presente trabalho focou apenas na análise de dados referentes ao estado do Rio Grande do Sul. Conforme mencionado anteriormente, apesar do RS alcançar a meta estabelecida de mortalidade infantil pelo PES, mais de 70% desses óbitos são neonatais (Secretaria da Saúde do Estado do Rio Grande do Sul, 2022).

Para o treinamento dos modelos, o conjunto de dados utilizado possui 1.132.701 registros, sendo 4.768 assinalados como óbitos neonatais, ou seja, aproximadamente 0,4% dos registros pertencem à classe positiva. O tamanho do conjunto de dados de teste é 265.338, onde 1.166 registros são da classe positiva, mantendo a mesma proporção das classes que o conjunto de dados de treinamento, como pode ser visto nos gráficos da

Figura 4.3.

Figura 4.3 – Distribuição das classes no conjunto de dados



Fonte: A autora

Para lidar com o desbalanceamento de classes, os seguintes métodos foram selecionados: adicionar pesos às classes através de uma abordagem sensível ao custo (se aplicável), subamostragem aleatória, SMOTE, SMOTETomek, SMOTEENN. Para o aprendizado, os seguintes algoritmos foram escolhidos: árvore de decisão, floresta aleatória, AdaBoost e XGBoost.

Primeiramente, para encontrar os melhores hiperparâmetros para cada algoritmo de aprendizado de máquina, foi utilizado GridSearchCV para cada um dos algoritmos selecionados, utilizando 10% do conjunto de dados de treinamento. No *grid search* foi utilizada a métrica *average precision* como função de perda, essa métrica calcula a precisão média de cada predição. Também foram permutados alguns possíveis valores para os hiperparâmetros de cada algoritmo, esses valores podem ser vistos com mais detalhes na tabela 4.3.

Tabela 4.3 – Grid Search e suas opções de hiperparâmetros

Algoritmo	**Possíveis Valores para os Hiperparâmetros**
Árvore de Decisão	*class_weight=['balanced', {0:0.9, 1: 0.1}, {0:0.99, 1: 0.01}, {0:0.8, 1: 0.2}]*
	max_depth=[None, 5, 10, 20]
	max_features=[“sqrt”, “log2”]
Floresta Aleatória	*class_weight=['balanced', {0:0.9, 1: 0.1}, {0:0.99, 1: 0.01}, {0:0.8, 1: 0.2}]*
	max_depth=[None, 5, 10, 20]
	max_features=[“sqrt”, “log2”]
	n_estimators=[100, 300, 500, 800, 1000]
AdaBoost	*n_estimators=[100, 300, 500, 800, 1000]*
XGBoost	*learning_rate=[0.1, 0.2, 0.3]*
	scale_pos_weight=[0.9, 0.99, 0.8]

A maioria dos hiperparâmetros foram configurados com os valores padrões de cada algoritmo, entretanto, a Tabela 4.4 cita os hiperparâmetros que foram configurados após etapa de otimização de hiperparâmetros com *grid search*.

Tabela 4.4 – Modelos e seus hiperparâmetros

Algoritmo	**Hiperparâmetros**
Árvore de Decisão	*class_weight={0:0.1, 1:0.9}, max_depth=5, max_features='sqrt'}*
Floresta Aleatória	*class_weight={0:0.1, 1:0.9}, max_depth=5, max_features='sqrt', n_estimators=1000*
AdaBoost	*n_estimators=100*
XGBoost	*learning_rate=0.3, scale_pos_weight=0.9*

Posteriormente à seleção dos hiperparâmetros, cada modelo foi treinado e testado, utilizando as diferentes estratégias para lidar com o desbalanceamento de classes. Ou seja, para cada algoritmo foram implementados modelos sem nenhum tratamento específico para desbalanceamento de classes, modelos com pesos nas classes (se aplicável), modelos com aplicação do método subamostragem aleatória, modelos com aplicação do método SMOTE, modelos com aplicação do método SMOTETomek e modelos com aplicação do método SMOTEENN. Também foram implementados modelos com dois algoritmos modificados para lidar com desbalanceamento de dados, o RUSBoost (do inglês, *Random UnderSampling Boost*) e o Florestas Aleatórias Balanceadas(DURAHIM, 2016).

Para cada modelo gerado, as métricas acurácia, acurácia balanceada, precisão, sensibilidade, f1-score, AUROC e AUPRC foram computadas e analisadas (BATISTA et al., 2021). Por fim, para o modelo com melhor resultado foram extraídos os atributos com maiores influências utilizando o método SHAP.

5 RESULTADOS

O objetivo dos classificadores que foram modelados é predizer o risco de óbito neonatal dado um registro de nascimento, portanto é importante obter modelos que tenham uma baixa taxa de falsos negativos, ou seja, o erro de predizer erroneamente que um óbito não vai ocorrer é pior do que o contrário. Sendo assim, as métricas mais importantes para avaliar a qualidade dos modelos são sensibilidade (*recall*), f1-score e AUPRC.

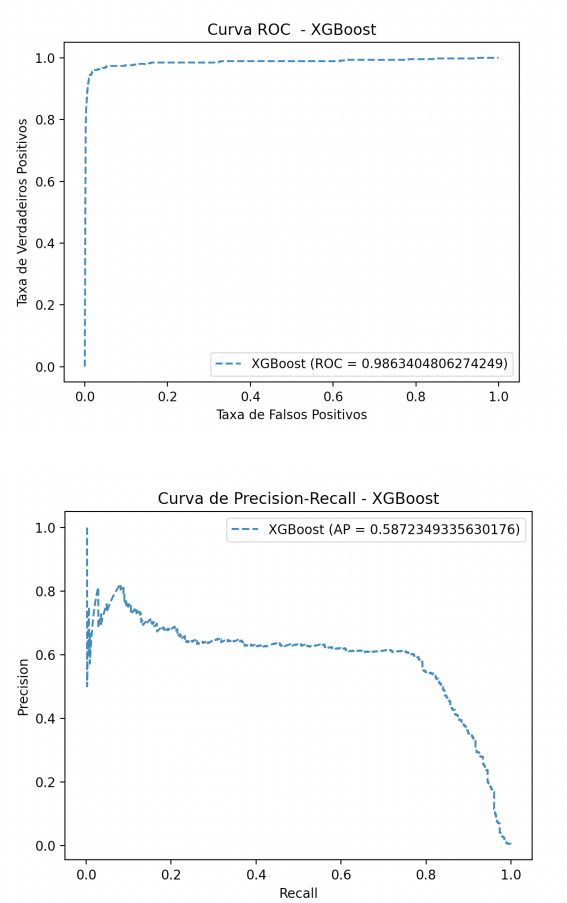
Observou-se no experimento que algoritmo que obteve o melhor desempenho para predizer possível óbito neonatal foi o XGBoost, mesmo sem nenhum método específico para lidar com desbalanceamento de classes. Esse modelo apresentou 26% de sensibilidade, 37% de f1-score e 57% de AUPRC. Porém, a combinação do método SMOTE-ENN com XGBoost obteve melhores resultados para as métricas de sensibilidade e f1-score, com 46% em cada; em contrapartida, o AUPRC diminuiu levemente para 42% neste modelo. As curvas ROC e *precision-recall* desse último modelo citado podem ser visualizadas com mais detalhes nas figuras 5.1 e 5.2.

Os modelos que utilizaram o algoritmo de árvore de decisão obtiveram os piores resultados, mesmo com os diferentes métodos de amostragem e abordagem sensível ao custo. Com esse classificador, a estratégia para lidar com desbalanceamento de classes que obteve melhores resultados foi a abordagem sensível ao custo, onde foi adicionado um peso $0,1$ a classe majoritária e o peso $0,9$ e apresentou 33% de precisão, 46% de sensibilidade, 38% de f1-score e 25% de AUPRC, apesar de métrica AUPRC ter diminuído 2% em relação ao modelo sem tratamento para desbalanceamento, a sensibilidade aumentou 26% em relação ao mesmo modelo. Os outros métodos combinados com esse classificador, aumentaram a sensibilidade em contrapartida diminuíram drasticamente a precisão dos modelos, ou seja, o número de falsos positivos aumentaram enquanto os falsos negativos diminuíram.

O modelo que foi implementado com o algoritmo de florestas aleatórias e pesos associados às classes obteve o melhor resultado dentre os modelos que empregaram o algoritmo de florestas aleatórias. O modelo em questão apresentou 36% de precisão, 52% de sensibilidade, 42% de f1-score e 37% de AUPRC, ou seja, esse modelo obteve resultados próximos ao modelo que combinou XGBoost e SMOTE-ENN, porém não obteve tão boa precisão quanto o modelo citado.

O modelo que foi implementado com o algoritmo de AdaBoost e SMOTE-ENN, obteve o melhor resultado dentre os modelos que empregaram o algoritmo de AdaBoost,

Figura 5.1 – Curvas de ROC e de *precision-recall* do modelo que utilizou o classificador XGBoost e método de amostragem SMOTE-ENN com dados de treino



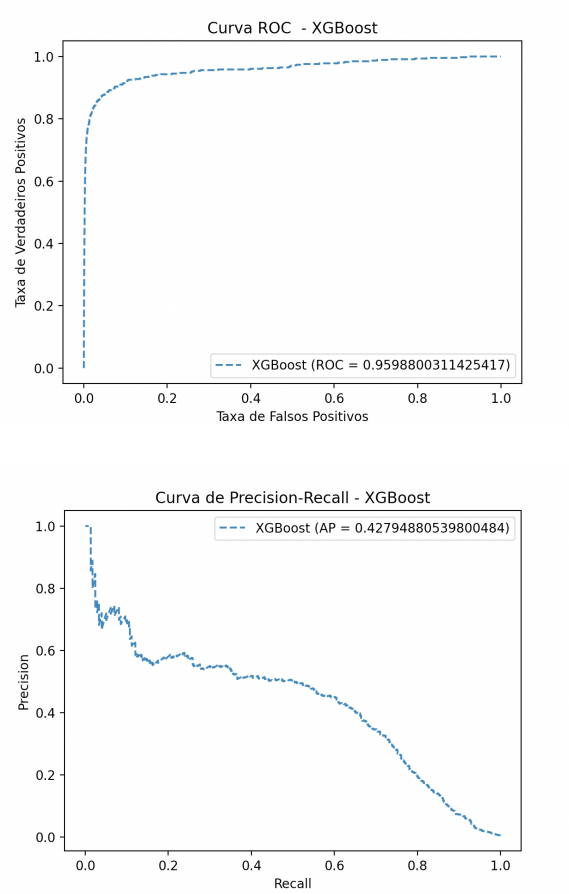
Fonte: A autora

pois o modelo em questão apresentou 34% de precisão, 55% de sensibilidade, 41% de f1-score e 26% de AUPRC, o que significa que o resultado não é distante do resultado do modelo XGBoost e SMOTE-ENN, porém é um resultado pior que o melhor modelo utilizando florestas aleatórias.

O método SMOTE-ENN, que é a técnica híbrida que combina o método de SMOTE e o algoritmo de limpeza ENN, apresentou melhora em todos os algoritmos. Essa estratégia demonstrou que é capaz de diminuir o *overfitting* que ocorre em problemas com classes desbalanceadas. Deve-se mencionar que o método SMOTE-Tomek obteve resultados bem próximos dos resultados obtidos pela técnica SMOTE-ENN.

O método de subamostragem aleatória apresentou resultados péssimos em todos os modelos. Nos resultados, pode-se observar que há *overfitting* nesses modelos, onde o modelo identifica a maioria dos casos como óbito neonatal. O método SMOTE indicou melhora na sensibilidade, porém diminuiu a precisão, os resultados ficaram próximos dos

Figura 5.2 – Curvas de ROC e de *precision-recall* do modelo que utilizou o classificador XGBoost e método de amostragem SMOTE-ENN com dados de teste



Fonte: A autora

resultados utilizando os métodos de amostragem híbridos. Os modelos implementados utilizando o RUSBoost (do inglês, *Random UnderSampling Boost*) e o Florestas Aleatórias Balanceadas apresentaram resultados parecidos com os modelos treinados com métodos de subamostragem.

O método de subamostragem apresentou resultados piores porque nesse tipo de método ocorre perda de importantes informações discriminativas disponíveis no conjunto de dados original, ressaltamos que o conjunto de dados original tem o tamanho de 1127933, com a reamostragem permaneceram 9.536 para serem treinados, ou seja, 1118397 dados da classe negativa foram descartados aleatoriamente.

Os métodos que utilizam SMOTE apresentaram melhores resultados, pois poucas informações relacionadas a classe positiva foram descartadas. O método SMOTE-ENN mostrou melhor resultado, pois nesse método é reduzido o risco de *overfitting* no classificador devido à introdução de exemplos artificiais na classe minoritária e a eliminação de

registros que não são muito importantes, com base na vizinhança. Embora o método de reamostragem tenha melhorado o desempenho do classificador de óbitos neonatais, ainda é um desafio conseguir alta sensibilidade do algoritmo sem perder muita precisão.

A Tabela 5.1 apresenta com mais detalhes os resultados obtidos com cada estratégia de modelagem realizada no experimento.

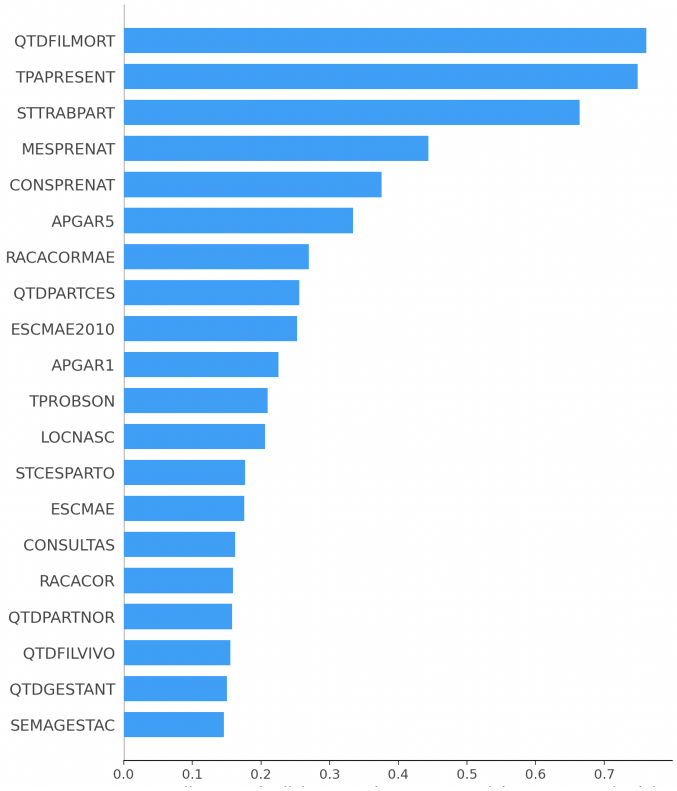
Tabela 5.1 – Resultados do experimento

Classificador	**Método de Amostragem/Peso**	**Acurácia**	**Acurácia Balanceada**	**Precisão**	**Sensibilidade**	**F1 Score**	**AUCROC**	**AUPRC**
Árvore de Decisão		1,00	0,56	0,70	0,13	0,22	0,85	0,27
Árvore de Decisão	Class Weights	0,99	0,73	0,33	0,46	0,38	0,88	0,25
Árvore de Decisão	Random Undersampling	0,05	0,52	0,00	0,99	0,01	0,88	0,07
Árvore de Decisão	SMOTE	0,95	0,84	0,06	0,73	0,11	0,87	0,18
Árvore de Decisão	SMOTE-Tomek	0,95	0,84	0,06	0,73	0,11	0,87	0,17
Árvore de Decisão	SMOTE-ENN	0,95	0,84	0,06	0,73	0,11	0,87	0,17
Florestas Aleatórias		1,00	0,54	0,76	0,07	0,14	0,93	0,39
Florestas Aleatórias	Class Weights	0,99	0,76	0,36	0,52	0,42	0,93	0,37
Florestas Aleatórias	Random Undersampling	0,13	0,56	0,00	0,99	0,01	0,92	0,24
Florestas Aleatórias Balanceadas	Class Weights	0,10	0,55	0,00	1,00	0,01	0,92	0,24
Florestas Aleatórias	SMOTE	0,95	0,88	0,07	0,81	0,12	0,92	0,23
Florestas Aleatórias	SMOTE-Tomek	0,95	0,88	0,07	0,81	0,12	0,92	0,23
Florestas Aleatórias	SMOTE-ENN	0,95	0,88	0,07	0,81	0,12	0,92	0,22
AdaBoost		1,00	0,61	0,57	0,23	0,33	0,93	0,29
AdaBoost	Random Undersampling	0,95	0,89	0,07	0,83	0,12	0,93	0,27
RUSBoostClassifier		0,84	0,78	0,02	0,72	0,04	0,83	0,12
AdaBoost	SMOTE	0,99	0,72	0,38	0,45	0,41	0,92	0,27
AdaBoost	SMOTE-Tomek	0,99	0,73	0,38	0,46	0,42	0,92	0,27
AdaBoost	SMOTEENN	0,99	0,76	0,34	0,55	0,41	0,92	0,26
XGBoost		1,00	0,63	0,64	0,26	0,37	0,96	0,57
XGBoost	Random Undersampling	0,99	0,84	0,20	0,70	0,31	0,96	0,36
XGBoost	SMOTE	1,00	0,64	0,54	0,28	0,37	0,94	0,44
XGBoost	SMOTE	1,00	0,63	0,57	0,26	0,36	0,95	0,50
XGBoost	SMOTE-Tomek	1,00	0,63	0,60	0,26	0,36	0,95	0,50
XGBoost	SMOTE-ENN	1,00	0,73	0,47	0,46	0,46	0,95	0,42

Os fatores que mais influenciaram a detecção de óbito neonatal no modelo que combinou a técnica de reamostragem SMOTE-ENN e o classificador XGBoost que foi o melhor modelo identificado neste trabalho foram a "quantidade de filhos nascidos mortos da mãe", "escala de *apgar* no quinto minuto", "mês no qual começou o pré-natal", "quantidade de partos normais", "se o trabalho de parto foi induzido"e "tipo de apresentação do recém nascido". A influência de outros fatores também são apresentados na Figura 5.3.

O estudo teve algumas limitações em relação a coleta dos dados, os dados foram coletados em diferentes sistemas e tiveram que serem unidos sem nenhum identificador único em comum entre as duas bases o que acarretou na perda de muitos registros de óbitos que não puderam ser utilizados, o que ocasionou um desbalanceamento de classes maior do que o realmente deveria ser. Uma outra limitação foi a quantidade original coletada de dados que demandou muito tempo de manipulação de dados, no entanto o conjunto original de dados não foi totalmente utilizado por limitações físicas para executar a aprendizagem com milhares de dados.

Figura 5.3 – Influência dos atributos nos modelos



Fonte: A autora

6 CONCLUSÃO

Esta pesquisa apresentou a comparação de metodologias para modelar classificadores que trabalham com problemas com alto desbalanceamento de classes, como é o caso de predizer o risco ou não de um óbito neonatal a partir de dados socieconômicos da mãe, da gestação e do bebê nascido vivo.

A partir de dados públicos coletados do governo brasileiro através do DATASUS, foi criado um novo conjunto de dados, compreendendo mais de 1 milhão de amostras para o problema da mortalidade neonatal no estado do Rio Grande do Sul. A redução da mortalidade infantil é ainda um desafio para os serviços de saúde e para a sociedade como um todo.

Nesse trabalho foram apresentado os desafios de modelar classificadores que lidam com problemas com classes severamente desbalanceadas, que é o caso da análise de mortalidade neonatal. As diferentes estratégias para trabalhar com desbalanceamentos de classes foram analisadas, sendo elas: métodos de subamostragem aleatória, SMOTE, SMOTE-ENN, SMOTE-Tomek, algoritmos ensemble modificados e aprendizado sensível ao custo. Nesse trabalho foi demonstrado que os modelos combinados com subamostragem aleatória e algoritmos modificados não auxiliam na performance de modelos com dados com desbalanceamento acentuados.

O experimento também concluiu que a adição de pesos nas classes melhorou o desempenho dos modelos com árvore de decisão e florestas aleatórias e que a técnica que utiliza SMOTE-ENN lida melhor com desbalanceamento de dados nos modelos que utilizam algoritmos de *boosting*, pois essa técnica diminui o perigo de ocorrer *overfitting* na modelagem.

Foi apresentado, através do modelo que combinou o método de reamostragem SMOTE-ENN e o classificador XGBoost, que os fatores que mais influenciaram a detecção de óbito neonatal foram a "quantidade de filhos nascidos mortos da mãe", "escala de apgar no primeiro minuto", "mês no qual começou o pré-natal", "quantidade de partos normais", "se o trabalho de parto foi induzido"e "tipo de apresentação do recém nascido".

Os resultados obtidos neste trabalho podem ser utilizados como base para modelagem de outros problemas da área da saúde com problema de desbalanceamento de classes, assim como podem ser extrapolados para investigação do poder preditivo de modelos e da eficácia de técnicas para tratar o desbalanceamento com a análise de óbitos neonatais para outros estados e regiões do Brasil.

Como trabalhos futuros, podem ser avaliados os efeitos da iniciativa hospital amigo da criança na taxa de mortalidade neonatal no Rio Grande do Sul, mas também podem ser analisados as mudanças de perfil pré e pós COVID-19 nos dados de mortalidade neonatal.

REFERÊNCIAS

ALBISUA, I. et al. The quest for the optimal class distribution: An approach for enhancing the effectiveness of learning via resampling methods for imbalanced data sets. **Progress in Artificial Intelligence**, v. 2, 2013. ISSN 21926360.

ALVES, L. C. et al. Assessing the performance of machine learning models to predict neonatal mortality risk in brazil, 2000-2016. **medRxiv**, 2020.

BATISTA, A. F. et al. Neonatal mortality prediction with routinely collected data: a machine learning approach. **BMC Pediatrics**, v. 21, 2021. ISSN 14712431.

BATISTA, G. E. A. P. A.; BAZZAN, A. L. C.; MONARD, M. C. Balancing training data for automated annotation of keywords: a case study. **In Proceedings of the Second Brazilian Workshop on Bioinformatics**, 2003.

BATISTA, G. E. A. P. A.; PRATI, R. C.; MONARD, M. C. A study of the behavior of several methods for balancing machine learning training data. **ACM SIGKDD Explorations Newsletter**, v. 6, 2004. ISSN 1931-0145.

BELUZO, C. E. et al. Machine learning to predict neonatal mortality using public health data from são paulo - brazil. **medRxiv**, 2020.

BREIMAN, L. Random forests. **Machine Learning**, v. 45, 2001. ISSN 08856125.

CHANG, Y. C.; CHANG, K. H.; WU, G. J. Application of extreme gradient boosting trees in the construction of credit risk assessment models for financial institutions. **Applied Soft Computing Journal**, v. 73, 2018. ISSN 15684946.

CHEN, C.; LIAW, A.; BRIEMAN, L. Using random forest to learn imbalanced data: Technical report no. 666. university of california, berkley. **Using Random Forest to Learn Imbalanced Data**, v. 110, 2004.

DORADO-DíAZ, P. I. et al. Applications of artificial intelligence in cardiology. the future is already here. **Revista Española de Cardiología (English Edition)**, v. 72, 2019. ISSN 18855857.

DU, W. et al. Building decision tree classifier on private data. **Proceedings of the IEEE international conference on Privacy, security and data mining-Volume 14**, 2002. ISSN 0034-4257.

DURAHIM, A. O. Comparison of sampling techniques for imbalanced learning. p. 181–191, 2016. ISSN 2148-3752.

EBENUWA, S. H. et al. Variance ranking attributes selection techniques for binary classification problem in imbalance data. **IEEE Access**, v. 7, 2019. ISSN 21693536.

EFRON, B.; TIBSHIRANI, R. Estimating the error rate of a prediction rule. **Journal of the American Statistical Association**, v. 78, 1983.

FERNáNDEZ, A. et al. Performance measures. In: **Learning from Imbalanced Data Sets**. Switzerland: Springer, 2018. chp. 3, p. 47–61.

FRANçA, E.; LANSKY, S. Anais do xvi encontro nacional de estudos populacionais. **Anais do XVI Encontro Nacional de Estudos Populacionais**, 2008. Available from Internet: <http://www.abep.org.br/publicacoes/index.php/anais/article/view/1763/1723http://www.abep.org.br/publicacoes/index.php/anais/article/view/1763/1723>.

GREENER, J. G. et al. **A guide to machine learning for biologists**. 2022.

Jhpiego Corporation World Health Organization. Making every baby count: Audit and review of stillbirths and neonatal deaths highlights from the world health organization 2016 audit guide. **WHO Library Cataloguing-in-Publication Data**, p. 1–4, 2016. ISSN 9241511222. Available from Internet: <http://www.who.int/about/www.mcsprogram.org>.

KOTSIANTIS, S. B.; ZAHARAKIS, I. D.; PINTELAS, P. E. Supervised machine learning : A review of classification techniques general issues of supervised learning algorithms. **Informatica (Ljubljana)**, v. 31, 2007. ISSN 03505596.

Laboratório de Estatística e Geoinformação. **Métodos de reamostragem**. 2023. Available from Internet: <http://cursos.leg.ufpr.br/ML4all/apoio/reamostragem.html>.

LUNDBERG et al. A unified approach to interpreting model predictions. In: GUYON, I. et al. (Ed.). **Advances in Neural Information Processing Systems**. Curran Associates, Inc., 2017. v. 30. Available from Internet: <https://proceedings.neurips.cc/paper_files/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf>.

MONARD, M. C.; BARANAUSKAS, J. A. Conceitos sobre aprendizado de máquina. In: **Sistemas Inteligentes Para Engenharia**. Belo Horizonte: Editora UFMG, 2003. chp. 4, p. 39–56.

Secretaria da Saúde do Estado do Rio Grande do Sul. **BOLETIM EPIDEMIOLÓGICO DO ESTADO DO RIO GRANDE DO SUL MORTALIDADE MATERNA, INFANTIL E FETAL 2022**. 2022. Available from Internet: <https://saude.rs.gov.br/upload/arquivos/202206/08164752-boletim-epidemiologico-sobre-mortalidade-materna-infantil-e-fetal-2022.pdf>.

Secretaria de Vigilância em Saúde. **Mortalidade infantil no Brasil**. 2021. Available from Internet: <https://www.gov.br/saude/pt-br/centrais-de-conteudo/publicacoes/boletins/boletins-epidemiologicos/edicoes/2021/boletim_epidemiologico_svs_37_v2.pdf>.

SEIFFERT, C. et al. Rusboost: A hybrid approach to alleviating class imbalance. **IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans**, v. 40, 2010. ISSN 10834427.

SáNCHEZ-HERNáNDEZ, F. et al. Predictive modeling of icu healthcare-associated infections from imbalanced data. using ensembles and a clustering-based undersampling approach. **Applied Sciences (Switzerland)**, v. 9, 2019. ISSN 20763417.

TRAN, T.; LE, U.; SHI, Y. An effective up-sampling approach for breast cancer prediction with imbalanced data: A machine learning model-based comparative analysis. **PLoS ONE**, Public Library of Science, v. 17, 5 2022. ISSN 19326203.

United Nations Children's Fund - UNICEF. **Committing to Child Survival : A Promise Renewed. Unicef Progress Report 2015**. [s.n.], 2015. Available from Internet: <https://www.who.int/pmnch/media/news/2012/promisebrochure.pdf>.

World Health Organization. **Newborn Mortality**. 2022. Available from Internet: <https://www.who.int/news-room/fact-sheets/detail/levels-and-trends-in-child-mortality-report-2021#>.