Modelando Classificadores Binários para Predição de AVC

Decio M. Filho[1]
Unicamp-IMECC,Campinas, SP

Resumo. Este estudo foca na predição de AVC(Acidente Vascular Cerebral), utilizando Regressão Logística e Random Forest. A relevância é destacada pela aplicação de técnicas avançadas de pré-processamento, ajuste de hiperparâmetros e otimização de thresholds. O desbalanceamento do conjunto de dados e a análise de características, como variáveis demográficas e hábitos de vida, são considerados. Exploramos métricas específicas, como Recall e AUC, ressaltando a importância de uma abordagem cuidadosa na escolha e avaliação de modelos. O estudo representa um esforço na busca por modelos precisos, adaptados às complexidades dos dados e desafios em saúde pública.

Palavras-chave. Dados Desbalanceados, Aprendizado de Máquina, AVC

1 Introdução

As doenças cardiovasculares representam uma preocupação global significativa, sendo responsáveis por uma parcela substancial de morbidade e mortalidade em todo o mundo[7]. Entre essas condições, o Acidente Vascular Cerebral (AVC). Nesse contexto, a previsão eficaz do risco de AVC torna-se crucial para possibilitar intervenções precoces e reduzir os impactos devastadores dessa condição.

Este estudo aborda especificamente a tarefa de predição de AVC, utilizando uma abordagem baseada em dois modelos distintos: **Regressão Logística e Random Forest**. A escolha desses modelos visa explorar diferentes estratégias de aprendizado de máquina para a construção de classificadores binários capazes de identificar pacientes propensos a sofrer um AVC.

O desafio apresentado pela natureza desbalanceada do conjunto de dados destaca a importância de estratégias específicas para lidar com esse desequilíbrio com as métricas mais apropriadas para tal.

2 Conjunto e Descrição dos Dados

Esse conjunto de dados foi retirado do **Kaggle** e aborda a previsão de AVC (Acidente Vascular Cerebral), a segunda principal causa de morte global, responsável por aproximadamente 11% de todas as mortes [4].O objetivo é prever se um paciente está propenso a ter um AVC com base em parâmetros como gênero, idade, diversas condições de saúde e hábitos de tabagismo. **Compostos por 5110 linhas e 12 variáveis**.

(Olhar mais informações no dataset - link disponível em considerações finais)

Tabela 1: Descrição das Variáveis do Conjunto de Dados

Variável	Tipo
id	int64
gender	object
age	float64
hypertension	int64
heart_disease	int64
ever_married	object
work_type	object
Residence_type	object
avg_glucose_level	float64
bmi	float64
smoking_status	object
stroke	int64

As 5110 linhas representam observações de pacientes que foram rotulados de de forma binária sobre terem tido ou não AVC, essa variável será a alvo a ser prevista pelo modelo.

[1]d236087@dac.unicamp.br

A variável id representa um identificador, não agregando nenhuma informação relevante para o modelo.

A variável **age** representa a idade dos indivíduos.

A variável **hypertension** ilustra se os pacientes tinham ou não hipertensão arterial.

A variável **heart disease** demonstra se os pacientes tinham ou não doneças cardíacas.

A variável **bmi** indica o índice de massa corpórea, métrica que indica relaçaõ entre o peso e altura - sendo bastante útil com problemas de saúde [8] Sobre as variáveis**gender** representa o gênero do paciente. Os valores possíveis são "Masculino", "Feminino"ou "Outro".

Já **ever married** indica se o paciente é casado ou não. Os valores possíveis são "Não"ou "Sim".

Work Type descreve o tipo de trabalho do paciente. Os valores possíveis incluem "Crianças", "Trabalho no governo", "Nunca trabalhou", "Privado"ou "Autônomo".

Residence Type indica se o paciente mora em uma área "Rural"ou "Urbana".

smoking status que descreve o status de fumante do paciente. Os valores possíveis são "Fumava anteriormente", "Nunca fumou", "Fuma"ou "Desconhecido".

Vale ressaltar que a variável target ('stroke') é desbalanceada, sendo composta de aproximadamente 95% da classe negativa (0) e 5% da positiva (1),sendo 0 não houve AVC e 1 como houve AVC, e que portanto precisaria de um conjunto de técnicas mais apropriada para esses desbalanceamento pois oferece um novo paradigma de dificuldade e exigências de tratamento [5]

3 Metodologia e Modelagem

3.1 Modelo de Regressão Logística

A regressão logística é uma técnica de aprendizado de máquina aplicada a problemas de classificação binária, onde um classificador $\psi : X \to C$ é determinado usando um conjunto de treinamento $T = (x_i, y_i) : i = 1, \ldots, m \subset X \times C$. O conjunto de classes C tem dois elementos, e a regressão logística é expressa como a composição de um regressor linear com a função logística.[10]

Formalmente, a função logística $f_\theta : \mathbb{R}^n \to [0, 1]$ é dada por $f_\theta(x) = \sigma \circ h_\theta(x)$, onde $\sigma(t) = \frac{1}{1+e^{-t}}$. A regressão logística é usada em problemas de classificação binária, interpretando $f_\theta(x)$ como a probabilidade estimada de x pertencer a uma das duas classes. Um classificador binário $\psi : \mathbb{R}^n \to 0, 1$ é definido com base na probabilidade estimada, onde $\psi(x) = 1$ se $f_\theta(x) \geq 0.5$ e $\psi(x) = 0$ caso contrário.[10]

O treinamento da regressão logística envolve a minimização da entropia binária cruzada em um conjunto de treinamento T. Apesar de não existir uma fórmula fechada para os parâmetros que minimizam a entropia binária cruzada, métodos de otimização convexa, como o gradiente descendente estocástico, podem ser utilizados.[10]

Além disso, é possível introduzir não-linearidade ao modelo aplicando transformações às características de entrada antes da regressão linear. A acurácia é frequentemente usada como métrica de desempenho para avaliar o classificador binário resultante.[10]

Em resumo, a regressão logística oferece uma abordagem flexível e interpretável para problemas de classificação binária, fornecendo uma probabilidade estimada para a pertinência a uma classe específica.

3.2 Modelagem de Random Forest

Uma Floresta Aleatória é um tipo de ensemble de árvores de decisão, onde cada árvore é treinada em um subconjunto diferente do conjunto de treinamento. O conceito central de ensembles é utilizar a sabedoria coletiva de múltiplos modelos para melhorar o desempenho preditivo. A previsão em um ensemble é obtida por meio de técnicas como média aritmética para regressão ou votação da maioria para classificação.[10]

As principais estratégias para diversificar as árvores de decisão em uma Floresta Aleatória incluem Bagging e Pasting, consistindo na divisão do conjunto de treinamento em subconjuntos distintos, amostrados com ou sem reposição. Isso reduz o viés e a variância dos modelos individuais. Além da divisão do conjunto de treinamento, é possível realizar uma seleção aleatória de características. Patches aleatórios combinam essa seleção com bagging, enquanto subespaços aleatórios usam todo o conjunto de treinamento com seleção aleatória de características. Ou seja,uma Floresta Aleatória é um ensemble de árvores de decisão treinadas com bagging e seleção aleatória de características nas divisões dos ramos.[10]

3.3 Pré-processamento dos Dados

Tratamento de Valores Faltantes: Identificação e preenchimento/exclusão de valores faltantes no conjunto de dados. Identificou-se apenas que na variável **bmi** haviam alguns valores faltantes, sendo excluídos do conjunto de dados.

Exclusão de Variáveis Desnecessárias: Remoção de variáveis irrelevantes para o modelo, como o identificador `id` que não fornece informações úteis.

Tratamento das Variáveis Numéricas e Categóricas:

Escalonamento de Variáveis Numéricas: Utilização da função `standard_scaler` da biblioteca `sklearn` para padronizar as variáveis numéricas. A padronização das variáveis é indicada para reduzir problemas numéricos e mehorar a performance, especialmente em algoritmos que sofre com o tamanho da escala [3]. Codificação de Variáveis Categóricas: Como o conjunto de dados possui grande parte de variáveis categóricas decidiu-se aplicar a codificação por Onehot Encoder para transformar as variáveis categóricas em formato adequado para o treinamento dos modelos

3.4 Estatística Descritiva

- Estatísticas Sumárias: Cálculou-se as estatísticas descritivas (média, desvio padrão, quartis, etc.) para entender a distribuição e tendências dos dados e não foi encontrada nenhuma anomalia que exigisse um tratamento mais robusto

- Análise de Correlação: Avaliação de correlações entre variáveis para identificar padrões e possíveis multicolinearidades. A multicolinearidade entre as variáveis preditoras de um modelo podem criar sérios problemas na estimação do modelo e afetar diretamente a eficiência desse modelo [2]. De acordo com a figura abaixo não foi identificada multicolinearidade significativa.

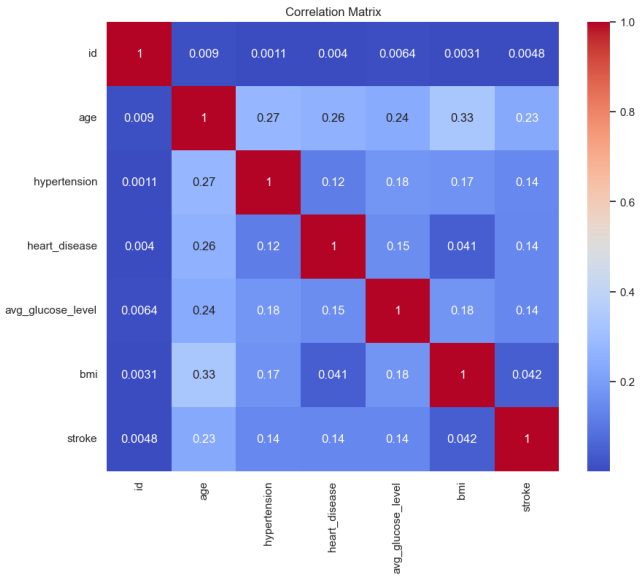

Figura 1: Matriz de Correlação

3.5 Divisão dos Dados

Separação em Conjuntos de Treino, Teste e Validação: Divisão dos dados em conjuntos de treino (60%), teste e validação (20% cada) para a construção e avaliação dos modelos.

3.6 Tunning de Hiperparâmetros

Para avaliação de hiperparâmetos dos modelos a serem treinados e avaliados utilizou-se o **Optuna**. O Optuna é um software de otimização de hiperparâmetros de próxima geração que atende a novos critérios de design. Ele possibilita aos usuários construir dinamicamente espaços de busca de parâmetros, implementar estratégias de busca e poda de forma eficiente, e ser facilmente implantado para diversos propósitos [1]

Para a **regressão logística**, um dos hiperparâmetros mais significativos é o parâmetro de regularização, frequentemente representado por C. A otimização eficaz de C desempenha um papel vital na capacidade do modelo de generalizar padrões a partir dos dados de treinamento para novas instância. Além da otimização de C, a utilização da Entropia Cruzada Ponderada (represetando pelo class_weight = balanced) como métrica de otimização é uma consideração valiosa. A Entropia Cruzada Ponderada leva em conta o desequilíbrio nas classes, atribuindo pesos diferentes às classes minoritárias e majoritárias. Isso é especialmente crucial quando a distribuição das classes não é uniforme, pois evita que o modelo seja tendencioso em direção à classe majoritária [9].

Para o modelo de **random-forest** n estimators: Refere-se ao número de árvores na floresta. No caso presente, o valor é 183, indicando um conjunto robusto de árvores que contribuem para a decisão final do modelo.

max depth: Este parâmetro determina a profundidade máxima de cada árvore na floresta. Aqui, o valor é 22, o que significa que cada árvore pode ter um máximo de 22 níveis de decisão.

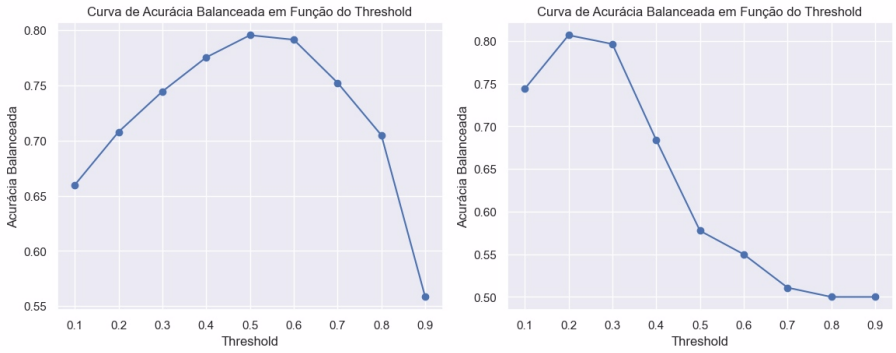

Figura 2: Otimização dos Limiares

min samples split: Indica o número mínimo de amostras necessárias para dividir um nó interno. O valor de 15 sugere que um nó só será dividido se houver pelo menos 15 amostras disponíveis.

min samples leaf: Representa o número mínimo de amostras exigido para ser uma folha (nó final) da árvore. Neste caso, são necessárias pelo menos 7 amostras para formar uma folha.

max features: Define o número máximo de características a serem consideradas ao procurar a melhor divisão. 'sqrt' indica que o número de características é a raiz quadrada do total, uma escolha comum para evitar superajuste.

class weight: Este parâmetro trata o desequilíbrio de classes. 'balanced' indica que as classes menos representadas receberão maior peso, contribuindo para uma abordagem mais equitativa na construção do modelo.

3.7 Mudança do Limiar de Decisão

A alteração do limiar de decisão em modelos de classificação pode ser uma estratégia benéfica, especialmente quando lidamos com conjuntos de dados desbalanceados. O limiar de decisão é o valor que determina a fronteira entre as classes preditas, e seu ajuste pode ter um impacto significativo em como o modelo lida com classes minoritárias e majoritárias.[6]

Com isso, fez se necessária a aplicação da curva ROC bem como uma métrica da análise acurácia balanceada em relação à variação de diferentes thresholds a fim de otimização. Além disso, fez-se uma mudança manual nas regiões ótimas a fim de investigar ainda mais um valor que se ajuste melhor à métrica proposta

Por conta disso, os limiares selecionados para os modelos de regressão logística e random forest foram 0.51 e 0.251, respectivamente.

4 Resultados

Nesta seção, apresentamos os resultados do treinamento e teste de ambos modelos de Regressão Logística e Random Forest. As métricas de avaliação utilizadas incluem Acurácia, Acurácia Balanceada, Recall e AUC (Área sob a Curva ROC). Acurácia vs. Acurácia Balanceada: Ao comparar a acurácia com a acurácia balanceada, observamos que a acurácia balanceada leva em consideração o desbalanceamento das classes. Em ambos os modelos, a acurácia balanceada é menor que a acurácia geral, indicando que o desbalanceamento impacta o desempenho.

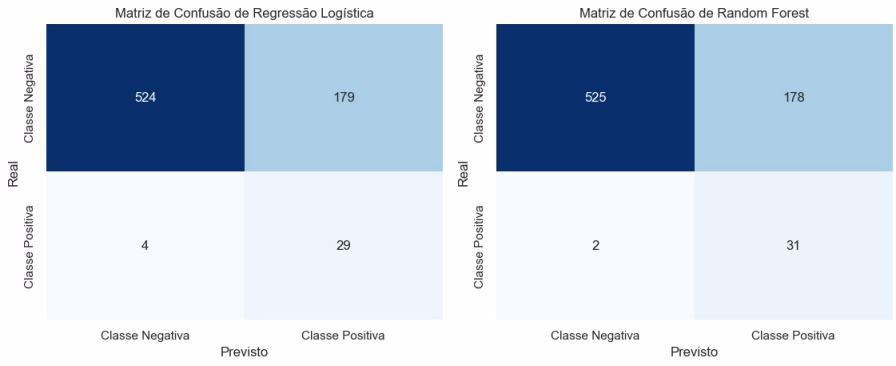

Figura 3: Matriz de Confusão dos Modelos

De acordo com a matriz de confusões,ambos os modelos têm resultados semelhantes em termos de acurácia geral, com a Random Forest apresentando uma ligeira vantagem. O Recall, que é crucial em problemas de saúde, é mais alta na Random Forest. Isso sugere que o modelo Random Forest é melhor em identificar corretamente os casos positivos (pessoas propensas a ter um AVC).

A escolha da métrica AUC é justificada pela sua invariância às mudanças de thresholds. Isso é particularmente importante em problemas de classificação, pois a escolha do threshold pode influenciar significativamente as métricas, como Recall.

Resultados da Comparação entre Regressão Logística e Random Forest

Métricas de Desempenho	**Regressão Logística**	**Random Forest**
Overall AUC	0.8593	0.8725
Balanced Accuracy	0.8121	0.8431
Accuracy	0.7514	0.7554
Recall	0.8788	0.9394

Sobre o recall, Observa-se que o modelo Random Forest apresenta um Recall mais alto (0.9394) em comparação com a Regressão Logística (0.8788). O Recall é crucial em problemas com classes desbalanceadas, especialmente em contextos de saúde, onde o objetivo é minimizar o custo dos falsos negativos. Nesses casos, é fundamental identificar corretamente os casos positivos, mesmo que isso resulte em um aumento nos falsos positivos.

Os thresholds utilizados para a decisão foram 0.51 para Regressão Logística e 0.252 para Random Forest.

Preliminarmente, testou-se os modelos sem grandes pré processamentos, apresentando métricas bem piores às apresentadas com a aplicação de todas as etapas ilustradas.

Esses resultados destacam a importância de considerar não apenas a acurácia geral, mas também métricas específicas, especialmente em problemas de saúde, onde o custo associado a falsos negativos pode ser substancial. O modelo Random Forest, com seu Recall mais alto, pode ser mais adequado em contextos onde a detecção eficaz de casos positivos é prioritária.

5 Conclusão

Neste estudo, abordamos a tarefa de previsão binária do risco de AVC utilizando dois modelos distintos: Regressão Logística e Random Forest. As etapas de pré-processamento e ajuste dos hiperparâmetros foram essenciais para melhor adaptação dos modelos, assim como o ajuste do limiar - crucial para dados desbalanceados. Em conclusão, a escolha entre os modelos deve levar em consideração a natureza específica do problema. Em contextos de saúde, onde a detecção eficaz de casos positivos é crucial, o modelo Random Forest pode ser mais adequado. A aplicação de técnicas de pré-processamento, otimização de hiperparâmetros e ajuste de limiares contribuiu significativamente para melhorar o desempenho dos modelos.

Os resultados obtidos destacam a importância de uma abordagem cuidadosa na escolha e avaliação de modelos para esse tipo de problema. O desbalanceamento de classes e a sensibilidade às classes positivas são aspectos críticos a serem considerados.

6 Considerações Finais

Os códigos gerados para toda a modelagem estão disponíveis no seguinte link:
`https://drive.google.com/drive/u/1/folders/1Is-BlEeshB2bZtZ91ykC5D87BMdCiqQT`

Os dados utilizados neste estudo foram obtidos diretamente do Kaggle e estão disponíveis em:
`https://www.kaggle.com/datasets/fedesoriano/stroke-prediction-dataset`

Agradecimentos

Agradeço ao professor Marcos Valle pelas excelentes aulas e pela dedicação em ajudar os alunos.

Referências

[1] Takuya Akiba et al. “Optuna: A Next-generation Hyperparameter Optimization Framework”. Em: **Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery Data Mining** (2019). DOI: 10.1145/3292500.3330701.

[2] A. Alin. “Multicollinearity”. Em: **Wiley Interdisciplinary Reviews: Computational Statistics** 2 (2011). DOI: 10.1002/wics.84.

[3] Timo Berthold e Gregor Hendel. “Learning To Scale Mixed-Integer Programs”. Em: (2021), pp. 3661–3668. DOI: 10.1609/aaai.v35i5.16482.

[4] J. Boldsen et al. “Better Diffusion Segmentation in Acute Ischemic Stroke Through Automatic Tree Learning Anomaly Segmentation”. Em: **Frontiers in Neuroinformatics** 12 (2018). DOI: 10.3389/fninf.2018.00021.

[5] Haibo He e E. A. Garcia. “Learning from Imbalanced Data”. Em: **IEEE Transactions on Knowledge and Data Engineering** 21 (2009), pp. 1263–1284. DOI: 10.1109/TKDE.2008.239.

[6] Justin M. Johnson e T. Khoshgoftaar. “Deep Learning and Thresholding with Class-Imbalanced Big Data”. Em: **2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)** (2019), pp. 755–762. DOI: 10.1109/ICMLA.2019.00134.

[7] M. Katan e A. Luft. “Global Burden of Stroke”. Em: **Seminars in Neurology** 38 (2018), pp. 208–211. DOI: 10.1055/s-0038-1649503.

[8] S. Kirk et al. “BMI: a vital sign for patients and health professionals.” Em: **The Canadian nurse** 105 1 (2009), pp. 25–8.

[9] Sheng Lu et al. “Dynamic Weighted Cross Entropy for Semantic Segmentation with Extremely Imbalanced Data”. Em: **2019 International Conference on Artificial Intelligence and Advanced Manufacturing (AIAM)** (2019), pp. 230–233. DOI: 10.1109/AIAM48774.2019.00053.

[10] **Notas de Aula - professor Marcos Eduardo Valle**. Disponível no Google Classroom de MS571.