APLICAÇÃO DE MODELOS DE MACHINE LEARNING VISANDO A PREDIÇÃO DA TRIAGEM DE PACIENTES

Pedro Augusto Azevedo Vilela[1] (IC), Edvard Martins De Oliveira[1] (PQ)
[1]*Universidade Federal de Itajubá - UNIFEI*

Palavras-chave: Machine learning. Mimics. Triagem de pacientes. Random forest. Xgboost.

Introdução

O aumento da demanda nos departamentos de emergência hospitalares gera a necessidade de ferramentas automatizadas que possam auxiliar na triagem rápida e precisa dos pacientes. A base de dados MIMIC-IV-ED contém informações sobre os atendimentos realizados nos departamentos de emergência, o que possibilita a criação de modelos preditivos para classificar a gravidade do estado clínico dos pacientes, otimizando o fluxo hospitalar. O objetivo deste trabalho é utilizar diferentes modelos de *machine learning* para prever a triagem dos pacientes e avaliar o desempenho de cada um deles. Os modelos avaliados incluem *Random Forest, XGBoost,* SVM, Regressão Logística, *Decision Tree,* e *TabNet,* sendo comparados em termos de acurácia e eficiência computacional em diferentes cenários de dados.

Metodologia

Foram utilizados dados da tabela *triage* do MIMIC-IV-ED, contendo informações sobre sinais vitais e características demográficas de aproximadamente 418 mil pacientes, além do nível de triagem selecionado para eles na admissão (JOHNSON et al., 2023). O processo de pré-processamento envolveu a exclusão de colunas textuais e a exclusão ou preenchimento pela média dos dados faltantes dependendo da situação. O problema de desbalanceamento de classes foi tratado através de técnicas de balanceamento para garantir uma distribuição equitativa entre as classes da coluna "*acuity*", que define qual o nível de triagem selecionado para aquele paciente.

Três cenários distintos foram avaliados:

1. Cenário 1: Dados completos sem balanceamento. Com um total de 418 mil amostras.
2. Cenário 2: Dados completos com balanceamento por pesos. Com um total de 418 mil amostras.
3. Cenário 3: Limitação do conjunto de dados a 10.000 amostras por classe. Com um total de 41 mil amostras.

No primeiro cenário, não houve nenhum tratamento de balanceamento na coluna "*acuity*" e todos os dados disponíveis na tabela foram utilizados no treinamento, resultando no número de amostras da figura 1.

```
acuity
3.0    225066
2.0    139411
4.0     28504
1.0     24019
5.0      1100
Name: count, dtype: int64
```

Figura 1 - Amostras para o cenário 1

No segundo cenário, houve um balanceamento por pesos em que as amostras que mais aparecem obtém um peso menor que as que menos aparecem, além de manter a utilização da totalidade dos dados disponíveis. Os pesos atribuídos a cada classe são mostrados na figura 2.

```
Classe 1: Peso 3.48
Classe 2: Peso 0.60
Classe 3: Peso 0.37
Classe 4: Peso 2.94
Classe 5: Peso 77.34
```

Figura 2 - Pesos para cada classe do cenário 2

No terceiro cenário, cada classe foi limitada a no máximo 10 mil amostras para realizar um balanceamento artificial. O número de amostras em cada classe está presente na figura 3, com o total de dados usados no treinamento sendo de 41.100 amostras devido a classe 5 possuir apenas 1.100 amostras. Não houve

nenhum outro tipo de balanceamento nesse caso.

```
acuity
2.0    10000
3.0    10000
4.0    10000
1.0    10000
5.0     1100
Name: count, dtype: int64
```

Figura 3 - Amostras para o cenário 3

Os modelos foram treinados e validados utilizando as métricas de acurácia e tempo de treinamento e com uma divisão de cada grupo de amostras em 80% dedicadas ao treinamento dos modelos e 20% para teste, com o objetivo de identificar o melhor modelo para cada cenário.

Resultados e discussão

No Cenário 1, mostrado na tabela 1, os melhores resultados foram alcançados pelos modelos *Random Forest* e *XGBoost*, com respectivas acurácias de 57% e 59% aproximadamente, enquanto o SVM mostrou desempenho inferior, com alto tempo de treinamento (10.756 segundos). O modelo *TabNet* também apresentou uma acurácia menor, possivelmente devido à sua complexidade em dados de tamanho moderado (Skiena, 2017).

Modelo	Acurácia	Tempo de Treinamento (s)
TabNet	0,4641	290.90
XGBoost	0,5922	4,18
Random Forest	0,5785	74,50
Logistic Regression	0,5539	3,57
SVM	0,5615	10.756,73
Decision Tree	0,4573	4,40

Tabela 1 - Resultados dos modelos para o cenário 1

No Cenário 2, mostrado na tabela 2, após o balanceamento dos dados, observou-se uma redução da acurácia no *XGBoost* para 18% e em todos os outros no geral, com apenas o *Random Forest* mantendo sua acurácia estável, indicando que o balanceamento impactou negativamente estes modelos. Esse fato é um comportamento anormal e não esperado, pois quando os dados passam por balanceamento, o esperado é as acurácias subirem. Tal acontecimento demanda maior análise e investigação em trabalhos futuros.

Modelo	Acurácia	Tempo de Treinamento (s)
TabNet	0,2411	372.52
XGBoost	0,1834	4,44
Random Forest	0,5814	82,10
Logistic Regression	0,2013	1,64
SVM	0,0627	143,50*
Decision Tree	0,4560	4,62

Tabela 2 - Resultados dos modelos para o cenário 2
* limitado a 1000 iterações

Já no Cenário 3, mostrado na tabela 3, com a limitação do número de amostras em cada classe, houve um aumento significativo na acurácia de todos os modelos. *Random Forest* e *XGBoost* obtiveram acurácia acima de 70%, com redução substancial no tempo de treinamento, tornando este cenário o mais eficiente em termos de custo-benefício. Entretanto, é importante salientar que o número de amostras reduzidas expõe as limitações dos modelos para o dataset completo. Também será melhor avaliado a possibilidade de *overfitting* nesse caso.

Modelo	Acurácia	Tempo de Treinamento (s)
TabNet	0,6704	23,43
XGBoost	0,7132	1,57
Random Forest	0,7112	5,20
Logistic Regression	0,5981	0,15
SVM	0,6543	35,93
Decision Tree	0,6086	0,27

Tabela 3 - Resultados dos modelos para o cenário 3

Conclusões

Os resultados indicam que os modelos *Random Forest* e *XGBoost* são os mais adequados para predição de triagem em bases de dados desbalanceadas, devido à sua alta acurácia e eficiência computacional (Skiena, 2017) (CHEN; GUESTRIN, 2016), enquanto os modelos *SVM* e *TabNet* apresentam tempo de treinamento muito superior aos outros testados, demonstrando seu grande custo computacional (Skiena, 2017) (ARIK; PFISTER, 2021). Os outros modelos testados não apresentaram destaque em acurácia, apenas em tempo de treinamento, o que viabiliza eles em casos em que tem-se o tempo como essencial ao invés da acurácia. O estudo também demonstrou a importância de técnicas adequadas de balanceamento e seleção de amostras para evitar degradação no desempenho dos modelos, entretanto alguns resultados anômalos foram encontrados quando o balanceamento foi feito, necessitando de maior investigação e análise dos possíveis motivos desse fato, além de mais estudos a fim de aumentar a acurácia dos modelos. Futuros trabalhos podem explorar técnicas avançadas de pré-processamento de dados, como aprendizado de representação, e a integração de outras tabelas do MIMIC-IV para melhorar os resultados preditivos.

Agradecimentos

Agradeço ao CNPq e a UNIFEI pelo incentivo financeiro e a oportunidade de aprofundar meus conhecimentos na área da computação através dessa pesquisa, de forma a me tornar um melhor estudante e, futuramente, um melhor profissional da área de engenharia.
Além disso, agradeço ao meu orientador, o Prof. Edvard Martins de Oliveira e o Prof. Mário Henrique Souza Pardo, que mostraram sua dedicação e empenho em me oferecer orientação acadêmica

Referências

JOHNSON, Alistair; BULGARELLI, Lucas; POLLARD, Tom; CELI, Leo Anthony; MARK, Roger; HORNG, Steven. MIMIC-IV-ED (version 2.2). PhysioNet, 2023. Disponível em: https://doi.org/10.13026/5ntk-km72. Acesso em: 26 set. 2024.

JOHNSON, Alistair E. W.; BULGARELLI, Lucas; POLLARD, Tom J.; CELI, Leo Anthony; MARK, Roger G.; HORNG, Steven. MIMIC-IV-ED Documentation. Version 2.2. PhysioNet, 2023. Disponível em: https://doi.org/10.13026/5ntk-km72. Acesso em: 26 set. 2024.

SKIENA, Steven S. *The data science design manual*. Cham: Springer, 2017. 445 p. (Texts in computer science). ISBN 9783319554440.

CHEN, Tianqi; GUESTRIN, Carlos. XGBoost: A Scalable Tree Boosting System. 2016. Disponível em: https://arxiv.org/abs/1603.02754. Acesso em: 26 set. 2024.

ARIK, Sercan; PFISTER, Tomas**.** TabNet: Attentive Interpretable Tabular Learning. 2021. Disponível em: https://arxiv.org/abs/1908.07442. Acesso em: 26 set. 2024.