

---

# MODELO DE INTELIGÊNCIA ARTIFICIAL PARA AUXILIAR NA TRIAGEM DE UNIDADES DE TERAPIA INTENSIVA



UM ESTUDO BASEADO NO PROTOCOLO DA AMIB

---

ALISON LUIS SASSI

UNIVERSIDADE REGIONAL DO NOROESTE DO  
ESTADO DO RIO GRANDE DO SUL

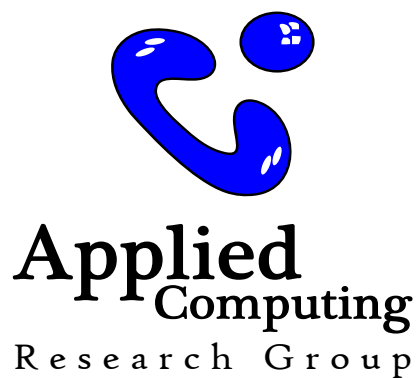
DISSERTAÇÃO DE MESTRADO

ORIENTADOR:

DR. RAFAEL ZANCAN FRANTZ

COORIENTADORA:

DRA. FABRICIA CARNEIRO ROOS FRANTZ



MARÇO, 2023

First published in March 2023 by  
Applied Computing Research Group - GCA  
Department of Exact Sciences and Engineering  
Rua Lulu Ilgenfritz, 480 - São Geraldo  
Ijuí, 98700-000, Brazil.

Copyright © MMXXII Applied Computing Research Group  
<http://www.gca.unijui.edu.br>  
[gca@unijui.edu.br](mailto:gca@unijui.edu.br)

In keeping with the traditional purpose of furthering science, education and research, it is the policy of the publisher, whenever possible, to permit non-commercial use and redistribution of the information contained in the documents whose copyright they own. You however are *not allowed* to take money for the distribution or use of these results except for a nominal charge for photocopying, sending copies, or whichever means you use redistribute them. The results in this document have been tested carefully, but they are not guaranteed for any particular purpose. The publisher or the holder of the copyright do not offer any warranties or representations, nor do they accept any liabilities with respect to them.

**Financiamento:** Bolsa de mestrado concedida pela CAPES.

Catálogo na Publicação

S252m

Sassi, Alison Luis.

Modelo de inteligência artificial para auxiliar na triagem de unidades de terapia intensiva / Alison Luis Sassi. – Ijuí, 2023.

125 f. : il. ; 30 cm.

Dissertação (mestrado) – Universidade Regional do Noroeste do Estado do Rio Grande do Sul (Campus Ijuí). Modelagem Matemática e Computacional.

“Orientador: Prof. Dr. Rafael Zancan Frantz”.

“Coorientadora: Profa. Dra. Fabrícia Carneiro Roos Frantz”.

1. Unidade de Terapia Intensiva (UTI). 2. Auxílio médico. 3. Inteligência artificial. 4. Gerenciamento hospitalar. 5. Protocolo de triagem. I. Frantz, Rafael Zancan. II. Frantz, Fabrícia Carneiro Roos. III. Título.

CDU: 007.52:616-08

Cristina Libert Wiedtkenper  
CRB10/2651

**UNIJUÍ - Universidade Regional do Noroeste do Estado do Rio Grande do Sul**  
**Programa de Pós-Graduação *Stricto Sensu* em Modelagem Matemática e Computacional**

A Comissão Examinadora, abaixo assinada, aprova a Dissertação

**Modelo de inteligência artificial para auxiliar na triagem de  
Unidades de Terapia Intensiva: Um estudo baseado no protocolo  
da AMIB**

Elaborada por

**Alison Luis Sassi**

Como requisito parcial para a obtenção do título de Mestre em Modelagem Matemática e  
Computacional

**Comissão Examinadora**

*Rafael Zancan Frantz*

Prof. Dr. Rafael Zancan Frantz  
(Orientador/UNIJUÍ)

*Fabricia Carneiro Roos Frantz*

Prof.<sup>a</sup> Dra Fabricia Carneiro Roos Frantz  
(Coorientadora/UNIJUÍ)

*Sandro Sawicki*

Prof. Dr. Sandro Sawicki  
(UNIJUÍ)

*José Joaquim Bocanegra García*

Prof. Dr. José Joaquim Bocanegra García  
(Universidad de Los Andes - Colômbia)

Ijuí, RS, 20 de abril de 2023.





*"Cada um contribua segundo propôs no seu coração, não com tristeza ou por necessidade; porque Deus ama ao que dá com alegria." 2 Coríntios 9:7*



*Dedico este trabalho a Deus, que sempre foi o autor da minha vida e do meu destino, a minha esposa, que me deu apoio nos momentos difíceis, ao meu orientador que me deu suporte em toda a trajetória como também ao José Salomão, amigo, motivador e que possibilitou esse trabalho.*



---

# Conteúdo

---

<b>Índice de abreviaturas</b> .....	<b>vii</b>
<b>Agradecimentos</b> .....	<b>ix</b>
<b>Resumo</b> .....	<b>xi</b>
<b>Abstract</b> .....	<b>xiii</b>
<b>1 Introdução</b> .....	<b>1</b>
1.1 Contexto da pesquisa .....	1
1.2 Motivação .....	2
1.3 Objetivos .....	3
1.4 Hipótese .....	4
1.5 Trabalhos relacionados .....	4
1.6 Estrutura do documento .....	7
<b>2 Materiais e Métodos</b> .....	<b>9</b>
2.1 Metodologia de Pesquisa .....	9
2.1.1 Método científico .....	9
2.1.2 Método de trabalho .....	11
2.2 Resumo do capítulo .....	17
<b>3 Referencial Teórico</b> .....	<b>19</b>
3.1 Fluxo e protocolo de triagem de pacientes na UTI .....	19
3.1.1 Fluxo do processo da admissão do paciente .....	20
3.1.2 Protocolo de alocação de recursos AMIB/ABRAMEDE ..	21
3.2 Inteligência Artificial .....	29
3.2.1 Árvore de decisão .....	34

3.2.2	Floresta aleatória .....	35
3.3	Resumo do capítulo .....	38
<b>4</b>	<b>Modelo Computacional Proposto .....</b>	<b>39</b>
4.1	Visão geral .....	39
4.2	Geração dos datasets .....	40
4.3	Sistema de validação de dados .....	49
4.4	Tecnologias utilizadas .....	51
4.5	Ambiente de análise de dados .....	52
4.6	Resumo do capítulo .....	55
<b>5</b>	<b>Aplicação do Modelo .....</b>	<b>57</b>
5.1	Descrição do experimento .....	57
5.2	Sistema de validação .....	57
5.3	Gerenciamento dos dados .....	63
5.4	Resultados e discussões .....	79
5.5	Dificuldades .....	91
5.6	Resumo do capítulo .....	91
<b>6</b>	<b>Considerações Finais .....</b>	<b>93</b>
6.1	Conclusão .....	93
6.2	Trabalhos futuros .....	95
	<b>Bibliografia .....</b>	<b>97</b>

---

## Índice de figuras

---

2.1	Fases do trabalho. ....	11
2.2	Processos de desenvolvimento. ....	15
3.1	Fluxo do modelo de <i>machine learning</i> . (Adaptado de Chappell [10], Souza [61]). ....	33
3.2	Representação de um <i>random forest</i> [1]. ....	37
4.1	Paciente por taxa de mortalidade. ....	46
4.2	Arquitetura do Django. ....	50
4.3	Arquitetura do sistema de validação proposto. ....	51
5.1	Tela de <i>login</i> do sistema de validação. ....	58
5.2	Tela de introdução ao sistema. ....	59
5.3	Tela da escolha dos grupos no sistema. ....	59
5.4	Tela de validação ao sistema. ....	60
5.5	Tela de justificativa do sistema. ....	60
5.6	Tela de agradecimento do sistema. ....	61
5.7	Amostra dos dados no arquivo CSV. ....	64
5.8	Gráfico da distribuição do treinamento dos dados. ....	80
5.9	Gráfico da distribuição dos dados da validação externa. ....	84
5.10	Estrutura de dados montada com a diferença e produto. ....	89



---

## *Índice de tabelas*

---

3.1	Citérios do protocolo de alocação de recursos AMIB/ABRAMEDE. (Adaptado de Conselho Federal de Medicina (CFM) [13]).	24
3.2	Pontuação SOFA por taxa de mortalidade.	25
3.3	Classificação do sistema neurológico.	26
3.4	Classificação do sistema respiratório.	26
3.5	Classificação do sistema cardiovascular.	27
3.6	Classificação do sistema de coagulação.	27
3.7	Classificação do sistema hepático.	28
3.8	Classificação do sistema renal.	28
4.1	Configurações do computador utilizado.	53
5.1	Visualização dos resultados das métricas.	74
5.2	Resultado da importância das variáveis.	81
5.3	Resultado das métricas do experimento.	82
5.4	Listagem comparativa entre médico e modelo.	86
5.5	Lista comparativa - médica vs modelo.	87



---

## *Índice de abreviaturas*

---

**ABRAMEDE** - Associação Brasileira de Medicina de Emergência

**AMIB** - Associação de Medicina Intensiva Brasileira

**ANCP** - Academia Nacional de Cuidados Paliativos

**CFM** - Conselho Federal de Medicina

**CSS** - Cascading Style Sheets

**CSV** - Comma-Separated Values

**CPU** - Central Processing Unit

**DCB** - Dilemas COVID-19 Bioética

**DSR** - Design Science Research

**EHR** - Registros Eletrônicos de Saúde

**GB** - Gigabyte

**GM** - Gabinete do Ministro

**HTML** - HyperText Markup Language

**IA** - Inteligência Artificial

**ML** - Machine Learning

**MPMII0** - Mortality Probability Model II

**MS** - Ministério da Saúde

**MTV** - Model Template View

**PLN** - Processamento de Linguagem Natural

**RNA** - Rede Neural Artificial

**SBGG** - Sociedade Brasileira de Geriatria e Gerontologia

**SOFA** - Sequential Organ Failure Assessment

**SQL** - Structured Query Language

**SUS** - Sistema Único de Saúde

**TB** - Terabytes

**UTI** - Unidade de Terapia Intensiva

---

## *Agradecimentos*

---

**A**gradeço a Deus pela oportunidade, capacidade de estar desenvolvendo esse trabalho em conjunto com pessoas especiais, meu orientador Dr. Rafael Z. Frantz, Dr. José Salomão Júnior e Dra. Isadora Salomão, que acreditaram e acompanharam do início ao fim. Assim como a CAPES, que acreditou e concedeu a bolsa de estudos, pois sem ela essa pesquisa não seria possível.

Agradecimento especial ao meu orientador que incentivou, orientou, motivou e esteve sempre presente para ajudar com os obstáculos da pesquisa.

Agradecimento especial ao José Salomão Júnior e sua esposa Isadora, que acreditaram na proposta, estiveram presentes durante a construção do trabalho e, com os quais construí uma grande amizade.

Ao amigo e colega Gustavo Spiess que auxiliou, sugeriu e acompanhou alguns momentos do trabalho, assim como os médicos voluntários envolvidos e que sempre estiveram dispostos a conversar e auxiliar na validação dos dados.

Por fim, agradecer a todos envolvidos no Programa de Pós-Graduação em Modelagem Matemática e Computacional que apoiaram e proporcionaram esse trabalho.



---

## Resumo

---

*O início de todas as coisas é pequeno.*

*Marcus T. Cicero, Filósofo Romano (106 AC - 43 AC)*

**A**s internações de pessoas em leitos de Unidade de Terapia Intensiva (UTI) aumentaram aceleradamente diante dos casos clínicos críticos suscitados pelo vírus COVID-19, ocasionando a escassez de leitos hospitalares, durante alguns meses. No cenário da escassez de leitos de UTI, o médico intensivista vivencia a inquietude de uma decisão da escolha do próximo paciente à medida que os leitos vão sendo desocupados. Essa situação pode causar problemas psicológicos, visto que a escolha pode ser influenciada por fatores emocionais do médico, bem como por influências políticas e sociais, o que pode entrar em conflito com a ética profissional e prejudicar indevidamente outro paciente com maior probabilidade de recuperação. Geralmente, a decisão de qual paciente deve ocupar um leito segue o conhecimento e experiência do médico encarregado, que utiliza formas de triagem que o auxiliam. A utilização de protocolos pode ajudar a padronizar as escolhas do médico, levando em conta o estado clínico dos pacientes. Entretanto, em certas situações, pode ocorrer empate na avaliação clínica de pacientes, o que pode dificultar a escolha de qual deles ocupará o leito hospitalar. Adicionalmente, é possível que fatores emocionais exerçam influência na decisão do médico, levando a escolhas inadequadas. Com um modelo computacional capturando as variáveis de decisão baseadas em dados do paciente, o profissional poderá se apoiar na transparência da sugestão que o modelo oferece. Para esse modelo baseado em dados do paciente se tornar factível, será necessário utilizar um protocolo que através de variáveis informadas, possa fazer uma listagem dos pacientes que necessitarão de um leito de UTI. O sistema utiliza modelo de Inteligência Artificial supervisionado, treinado para que o médico visualize os dados do paciente e sua respectiva prioridade dentre os que necessitam de internação na UTI.

**Palavras-chaves:** Unidade de Terapia Intensiva (UTI); Auxílio médico; Inteligência Artificial; Gerenciamento Hospitalar; Protocolo de Triagem.



---

# Abstract

---

*The beginnings of all things are small.*

*Marcus T. Cicero, Roman philosopher (106 BC - 43 BC)*

**T**he admission of people to Intensive Care Unit (ICU) beds has increased rapidly due to the critical clinical cases raised by the COVID-19 virus, causing a shortage of hospital beds for some months. In the scenario of ICU beds shortage, the intensivist physician experiences the anxiety of a decision to choose the next patient as the beds are being vacated. This situation may cause psychological problems, since the choice may be influenced by the physician's emotional factors, as well as by political and social influences, which may conflict with professional ethics and unduly harm another patient with a greater chance of recovery. Generally, the decision of which patient should occupy a bed follows the knowledge and experience of the physician in charge, who uses forms of triage to assist him or her. The use of protocols can help standardize the doctor's choices, taking into account the clinical condition of the patients. However, in certain situations, there may be a tie in the clinical evaluation of patients, which can make it difficult to choose which one will occupy the hospital bed. Additionally, it is possible that emotional factors may influence the physician's decision, leading to inappropriate choices. With a computer model capturing the decision variables based on patient data, the practitioner can rely on the transparency of the suggestion that the model offers. For this model based on patient data to become feasible, it will be necessary to use a protocol that through informed variables, can list the patients that will need an ICU bed. The system uses a supervised Artificial Intelligence model, trained so that the physician can visualize the patient's data and its respective priority among those who need ICU admission.

**Keywords:** Intensive Care Unit (ICU); Medical assistance; Artificial intelligence; Hospital Management; Triage Protocol.



---

# Capítulo 1

## Introdução

---

*The journey of a thousand miles  
begins with a single step.*

*Chinese proverb*

### 1.1 Contexto da pesquisa

A pandemia de COVID-19 trouxe uma escassez significativa de recursos, que obrigou as organizações de saúde a adaptar e criar protocolos de triagem mais eficientes para gerenciar essa escassez. Um exemplo disso é o Protocolo AMIB/ABRAMEDE, que foi atualizado recentemente para se adequar às necessidades atuais de gerenciamento de recursos durante a pandemia. Esse protocolo é amplamente utilizado por hospitais e outras instituições de saúde em todo o país para garantir que os recursos disponíveis sejam alocados de maneira justa e ética [15].

Durante a pandemia de COVID-19, a escassez de recursos pode suscitar questionamentos éticos sobre a alocação desses recursos. As organizações de saúde recomendam a implementação de protocolos de triagem, a fim de minimizar o número de mortes causadas pela doença e garantir o uso adequado dos recursos disponíveis. A falta desses protocolos pode resultar em uma distribuição injusta ou aleatória dos recursos escassos, o que pode ter consequências graves para os pacientes que dependem desses recursos para sobreviver.

Para evitar a alocação subjetiva e inconsistente de recursos escassos durante a pandemia de COVID-19, a Associação de Medicina Intensiva Brasileira (AMIB) publicou um protocolo em 2020 que fornece regras éticas

defensáveis para garantir que esses processos sejam realizados de maneira clara, transparente e tecnicamente bem fundamentada. Esse protocolo também garante que as escolhas médicas estejam de acordo com as leis brasileiras e sejam justificadas eticamente, evitando que essas decisões sejam tomadas em segredo ou sem um registro adequado [31].

## 1.2 Motivação

Em situações em que todos os leitos de UTI de um hospital estão ocupados e há pacientes esperando por uma vaga, é necessário que o profissional responsável pela UTI adote uma abordagem criteriosa para determinar qual paciente será alocado a um leito subsequentemente, quando houver disponibilidade. Durante esse processo, o médico pode ser influenciado por fatores como estresse, tensão emocional e fadiga física, que podem afetar sua avaliação e decisão. Isso cria uma situação desafiadora, especialmente em pandemias, como a COVID-19, em que a demanda por leitos de UTI tende a ser muito alta [43].

Uma das soluções possíveis para minimizar esse problema é a utilização de inteligência artificial (IA), que permite ensinar ao sistema a tomar decisões baseadas em variáveis bem definidas. Para isso, propomos um modelo computacional que utiliza os dados de saúde de cada paciente para avaliar a gravidade de cada caso e tomar decisões sobre a prioridade de cada paciente para o acesso aos leitos de UTI. Dessa forma, a IA pode ajudar a aliviar a carga de trabalho dos profissionais de saúde, auxiliando-os na tomada de decisões e permitindo que eles se concentrem no atendimento direto aos pacientes.

O modelo computacional proposto nesta pesquisa está baseado em recomendações de triagem criado e aperfeiçoado pela Associação de Medicina Intensiva Brasileira (AMIB) e Associação Brasileira de Medicina de Emergência (ABRAMEDE) em 2020. A recomendação AMIB/ABRAMEDE foi validado por outras associações médicas do Brasil, o que indica confiabilidade e eficácia. A recomendação consiste em um conjunto de regras que objetivam auxiliar os profissionais da medicina na avaliação da gravidade do estado de saúde dos pacientes, a fim de embasar a tomada de decisão quanto à priorização de ingresso em leitos de UTI [31]. O modelo computacional proposto utiliza as regras da recomendação para ajudar a determinar a ordem em que os pacientes devem ocupar os leitos disponíveis, levando em consideração informações sobre o estado de saúde de cada paciente.

Esta pesquisa propõem o desenvolvimento de uma ferramenta de IA que visa auxiliar os médicos intensivistas no gerenciamento de leitos de UTI

escassos, disponibilizando uma listagem ordenada de pacientes que estão esperando por um leito. O desenvolvimento desta ferramenta proporcionará transparência ao médico no acesso à listagem de pacientes elegíveis para UTI. Isso permitirá que os resultados sejam analisados, juntamente com a opinião do médico, facilitando a tomada de decisão e agilizando o processo de atribuição de vagas para pacientes com necessidade iminente de internação em UTI. A utilização da IA proporciona uma abordagem objetiva e livre de influências emocionais na seleção de pacientes, uma vez que ela se baseia em dados do paciente e nas recomendações da AMIB/ABRAMEDE de alocação de recursos em esgotamento durante a pandemia por COVID-19. Isso garante que a ferramenta considere informações relevantes sobre o estado de saúde dos pacientes, auxiliando o médico intensivista a focar no tratamento e no cuidado mais adequado, de acordo com as necessidades individuais do paciente.

## 1.3 **Objetivos**

Nesta seção, são apresentados o objetivo geral e os objetivos específicos que guiarão o desenvolvimento desta dissertação, com o intuito de aprofundar o conhecimento na área de estudo em questão.

### **Geral**

*Desenvolver um modelo computacional fundamentado em inteligência artificial que seja capaz de analisar dados de pacientes e indicar aquele paciente com maior necessidade de ingresso em um leito de UTI.*

### **Específicos**

- Extrair os critérios necessários do protocolo AMIB/ABRAMEDE para produção do modelo computacional.
- Construir bases de dados a partir dos critérios extraídos, garantindo transparência e confiabilidade.
- Desenvolver um *software* para validação médica de dados seletos.
- Propor aplicar e validar o algoritmo de aprendizado de máquina supervisionado capaz de interpretar os critérios.
- Validar os resultados obtidos.

## 1.4 Hipótese

Por meio da combinação de algoritmos de Inteligência Artificial e modelagem matemática, é possível criar um modelo computacional que recomenda de forma mais transparente e imparcial os pacientes que devem ter prioridade para a ingressarem em leitos de UTI, a fim de apoiar os profissionais de saúde em decisões críticas.

## 1.5 Trabalhos relacionados

Nesta seção, realiza-se uma revisão sistemática da literatura técnica e científica com o propósito de identificar estudos relevantes que abordem diretamente ou tangencialmente o tema de pesquisa em questão.

### Artigos acadêmicos

O trabalho de Feng et al. [21] descreve um sistema de triagem de IA desenvolvido para prever a sobrevivência e a utilização de recursos em pacientes gravemente doentes com COVID-19. O sistema foi treinado com dados de mais de 13.000 pacientes hospitalizados com COVID-19 e foi capaz de prever a sobrevivência e a utilização de recursos com precisão acima de 80%. Além disso, o sistema foi capaz de identificar padrões nos dados que foram relevantes para a previsão de sobrevivência e utilização de recursos, incluindo fatores como idade, gênero e comorbidades. Os autores concluem que o sistema de triagem de IA pode ser útil para ajudar a gerenciar a carga de trabalho em hospitais durante a pandemia de COVID-19, fornecendo informações precisas sobre a probabilidade de sobrevivência e a utilização de recursos em pacientes gravemente doentes. No entanto, eles também ressaltam a importância de levar em consideração outros fatores além dos dados médicos, como preferências pessoais e considerações éticas, na tomada de decisões sobre o tratamento de pacientes com COVID-19.

Com o objetivo de descrever as características clínicas e determinar fatores que predizem a admissão à unidade de terapia intensiva (UTI) de pacientes com COVID-19, Izquierdo et al. [27] desenvolveu um trabalho cujo resultado faz combinação de métodos epidemiológicos clássicos, processamento de linguagem natural (PLN) e aprendizado de máquina (para modelagem preditiva) para analisar os registros eletrônicos de saúde (EHRs) de pacientes com COVID-19. Essa combinação de métodos pôde ser analisada e

utilizada para este trabalho. A descrição dos resultados evidencia que é possível que uma combinação de variáveis clínicas de fácil obtenção prediga se os pacientes com COVID-19 necessitariam de internação na UTI.

O estudo de Kogien et al. [29], realizado em 2017 visa investigar a aplicação da ferramenta *Kanban* e da estratégia *just-in-time* (JIT) adaptado com a ficha de identificação do paciente, na gestão do tempo de permanência de pacientes em uma unidade de terapia intensiva (UTI) de um hospital público no Rio de Janeiro. Os autores relataram que a implantação da ferramenta *Kanban* e da estratégia JIT resultou em uma redução significativa do tempo de permanência de pacientes na UTI, algo semelhante à esta pesquisa em questão, bem como em uma melhoria na qualidade do atendimento e na eficiência dos processos. Adicionalmente, os autores propõem a implementação dessas técnicas para outras instituições de saúde, devido à sua capacidade de otimizar os recursos e melhorar a eficiência no uso de recursos limitados.

Côrtes et al. [14] examinam a utilização da idade como critério de priorização para leitos de UTI na primeira versão do protocolo de alocação de recursos em esgotamento durante a pandemia de COVID-19 criado pela ABRAMEDE e AMIB, comparando-o com a segunda versão que ajusta essa estratégia. O protocolo, proposto pelos autores em 2020, tem como objetivo auxiliar na tomada de decisões médicas em tempos de escassez de recursos hospitalares e prevê a priorização de leitos de UTI para pacientes com mais de 60 anos. Os autores questionam se essa estratégia é adequada sob o ponto de vista do princípio da dignidade da pessoa humana, pois ela pode levar à discriminação de pacientes mais jovens. Com base na análise dos dados publicados pelo governo da Bahia, os autores propõem a necessidade de garantir a isonomia para todos, conforme o Artigo 5º da Constituição Federal. Eles sugerem que, na segunda versão do protocolo, o critério de idade seja removido para torná-lo mais compatível com os direitos fundamentais dos idosos. No entanto, a questão ainda está em discussão entre as áreas técnicas e jurídicas para orientar as decisões médicas.

No artigo de Dall’Agnol [19], levanta-se uma análise comparativa da primeira com a segunda versão do Protocolo AMIB de alocação de recursos em esgotamento durante a pandemia por COVID-19. A versão inicial foi criada por um grupo de médicos da AMIB e, após a publicação, houveram reuniões que levaram a uma atualização publicada como uma segunda versão validada pelas instituições ABRAMEDE, SBGG e ANPC, a principal mudança foi a inclusão da medida de funcionalidade (ECOG) no lugar do critério de idade do paciente. O artigo destaca a diferença das versões na visão do grupo Dilemas COVID-19 Bioética (DCB), traz, também, uma situação de escolha entre

um pai idoso e o filho paralítico, a qual possui empate de pontuação do protocolo AMIB/ABRAMEDE. Conclui-se então que há uma discriminação na primeira versão do protocolo referente à idade (algo identificado pelas associações citadas) porém também há uma discriminação para pessoas com problemas de locomoção (o filho parapléxico) pois o ECOG é uma variável importante para o pontuação. Durante a elaboração do artigo, foi realizada uma entrevista com uma professora e médica para obter sua opinião sobre uma situação dilemática. A entrevistada afirmou que, em caso de escolha, ela optaria pela segunda versão, mas descartaria o critério ECOG a fim de basear sua decisão no julgamento clínico. Para a pesquisa em questão, o modelo computacional não contemplará casos de paraplegia ou situações atípicas, sendo essa determinação deixada a cargo da equipe clínica.

## Trabalhos técnicos

A tese de doutorado de Ramos [52] e a presente pesquisa compartilham objetivos semelhantes, porém a tese possui um enfoque mais específico na análise de uma ferramenta validada em um hospital. A tese possui o objetivo de avaliar o desempenho de um instrumento de auxílio à tomada de decisão para a priorização de vagas para doentes críticos em Unidades de Terapia Intensiva (UTIs) de um hospital acadêmico terciário em São Paulo. Essa avaliação se deu por variáveis clínicas, demográficas e administrativas, incluindo as variáveis necessárias para o cálculo de pontuação *Mortality Probability Model II* (MPMII0). Com sua conclusão sendo a construção e validação de um algoritmo de classificação de prioridades para admissão na UTI, que evidenciou boa reprodutibilidade, com uma concordância substancial entre pares e se correlacionando com as avaliações realizadas por médicos *experts* e a referência estabelecida [52].

A tese de doutorado de Cappello [9] realiza uma análise sob o aspecto bioético, jurídico e técnico dos critérios propostos pelo AMIB/ABRAMEDE. Foram levantados 251 prontuários de pacientes com diagnóstico da doença COVID-19, e que sofreram com a escassez de recursos em leitos de UTI, tornando uma pesquisa coorte retrospectiva de natureza quantitativa descritiva, estudos de coorte são aqueles que se analisa a partir de registros do passado até a data presente. Segundo o autor é esperado que em um modelo preditivo, haja equilíbrio entre a especificidade e sensibilidade, portanto, com a análise estatística pode-se demonstrar a capacidade preditiva do protocolo AMIB/ABRAMEDE sendo especificidade de 87% e sensibilidade de 58%, ou seja, bom desempenho para prever pacientes que podem sobreviver e menor desempenho para prever pacientes que podem vir a óbito, acurácia

sendo de 74% das amostras estudadas. O autor apresenta um comparativo com o protocolo SOFA isoladamente, que apresentou a especificidade de 64% e a sensibilidade de 79%, ou seja, é inversamente proporcional com o protocolo AMIB/ABRAMEDE. O autor ressalta em sua conclusão que deve-se expandir o trabalho com uma quantidade maior de amostras, obtendo um resultado melhor. Esta tese ressalta a importância do protocolo AMIB/ABRAMEDE também em seu aspecto jurídico, embasando a utilização desse protocolo para a atual pesquisa [9].

## 1.6 Estrutura do documento

Esta dissertação apresenta uma estrutura organizacional que segue a seguinte disposição:

**Capítulo 1** . Apresenta uma introdução a presente pesquisa, enquanto a Seção 1.1 é descrito o contexto da pesquisa, fornecendo uma compreensão mais profunda dos problemas e questões enfrentados na área de estudo; a Seção 1.2 apresenta a principal motivação que levou à realização desta pesquisa; a Seção 1.3 aborda o objetivo geral e lista os objetivos específicos; a Seção 1.4 apresenta a hipótese levantada pelo projeto; por fim, a Seção 1.5 discute trabalhos relacionados a esta pesquisa, oferecendo uma visão mais ampla das pesquisas já realizados na área.

**Capítulo 2**. Metodologias adotadas para a realização da pesquisa, no que se refere às estratégias e técnicas utilizadas para conduzir o estudo, descrita na Seção 2.1. A metodologia científica compreende a descrição das técnicas e abordagens teóricas adotadas para conduzir o estudo, enquanto a metodologia de trabalho apresenta os passos concretos conduzidos implementados para alcançar os objetivos definidos.

**Capítulo 3**. Proporciona uma revisão da literatura técnica e científica relacionadas à pesquisa desenvolvida nessa dissertação. A Seção 3.1 apresenta uma descrição detalhada da Unidade de Terapia Intensiva (UTI) e seu processo de triagem de pacientes durante condições de escassez de recursos, como também descreve o Protocolo de Alocação de Recursos e seus critérios; a Seção 3.2 descreve conceitos da inteligência artificial e suas subáreas juntamente com suas aplicações, como também algoritmos que foram utilizados; por fim a Seção 3.3 resume as informações apresentadas no capítulo.

**Capítulo 4**. Neste capítulo, são descritas as etapas de construção do artefato. Primeiramente, é apresentada uma visão geral dos processos na

Seção 4.1. Em seguida, a massa de dados é gerada utilizando o protocolo base, conforme detalhado na Seção 4.2. A arquitetura do software é descrita na Seção 4.3, que visa à validação dos dados gerados por médicos selecionados. A Seção 4.4 explana os motivos da escolha das tecnologias adotadas no estudo, enquanto a Seção 4.5 apresenta as ferramentas e bibliotecas utilizadas para a análise dos dados. Por fim, a Seção 4.6 resume as informações apresentadas neste capítulo.

**Capítulo 5.** Neste capítulo são apresentados os resultados da aplicação desenvolvida; a Seção 5.1 se refere à descrição das etapas do processo do experimento nesta pesquisa; a Seção 5.2 apresenta o artefato, o sistema para a validação médica, interface do sistema com os médicos; a Seção 5.3 demonstra a forma que foi criada para a geração de dados complementares aos dos médicos para que o treinamento seja mais assertivo; a Seção 5.4 realiza uma descrição e discussão dos resultados obtidos; a Seção 5.5 relata a dificuldade da validação médica dos dados; por fim, a Seção 5.6 Resume de forma geral o que foi tratado neste capítulo.

**Capítulo 6.** As considerações finais desta pesquisa são apresentadas neste capítulo. A Seção 6.1 discute as principais conclusões obtidas a partir do estudo realizado, enquanto a Seção 6.2 aborda as possíveis direções de pesquisa a serem exploradas a partir dos resultados encontrados.

---

# Capítulo 2

## Materiais e Métodos

---

*The journey of a thousand miles  
begins with a single step.*

*Chinese proverb*

**E**ste capítulo descreve os materiais e métodos utilizados nesta pesquisa. Inicialmente, é discutida a metodologia empregada para conduzi-la, seguida pela apresentação do protocolo de triagem, bem como das regras e critérios adotados para sua aplicação.

### 2.1 Metodologia de Pesquisa

A metodologia científica se refere às estratégias e técnicas utilizadas para conduzir o estudo, enquanto a metodologia de trabalho descreve os procedimentos práticos adotados para alcançar os objetivos estabelecidos. A escolha criteriosa e a aplicação rigorosa dessas metodologias são fundamentais para garantir a qualidade e a confiabilidade dos resultados obtidos, bem como para permitir sua generalização para outras situações similares. Assim, os esforços foram dedicados à seleção e aplicação adequada dessas metodologias, com vistas a assegurar a robustez do estudo e contribuir para o avanço do conhecimento científico.

#### 2.1.1 Método científico

A escolha do método denominado *Design Science Research* (DSR) para apresentar a solução de maneira científica se justifica pelo fato de que ele

coincide com o objetivo da pesquisa. Esse método científico se preocupa com a criação de artefatos informacionais eficientes, baseados em hipóteses teóricas fundamentadas no estado atual da arte e da técnica. O processo é rigoroso e visa produzir conhecimento científico de qualidade. Na utilização do método, é possível garantir que a solução apresentada esteja embasada em teorias sólidas e esteja de acordo com os padrões da ciência [49].

O DSR pode ser aplicado em uma ampla variedade de disciplinas, incluindo a área da saúde que enfrenta muitos desafios práticos e teóricos, incluindo a necessidade de encontrar soluções para problemas de saúde pública, como doenças infecciosas e crônicas, e para problemas relacionados ao atendimento médico, como a falta de acesso a cuidados de saúde de qualidade [18].

Um dos principais expoentes do DSR foi Herbert A. Simon, um cientista da computação e economista americano que ganhou o Prêmio Nobel de Economia em 1978. Em sua obra "*The Sciences of the Artificial*", Simon defendeu a ideia de que a ciência do projeto, ou seja, a ciência de projetar soluções para problemas reais, era uma forma importante de promover o avanço da ciência e da tecnologia [59].

Os primeiros autores a utilizar o termo *Design Science Research* foram Fuller (1965) e Gregory (1966), possuíam interesses parecidos, sendo a necessidade de buscar formas sistemáticas para melhorar e projetar artefatos, surgindo então o termo *Design Science Research* também conhecida como pesquisa baseada em *Design Science* [53].

O ciclo do DSR em uma pesquisa, se concentra na criação de artefatos de design para resolver problemas práticos em contextos do mundo real. De acordo com Peffers et al. [48], o ciclo DSR é composto por seis etapas: (1) identificação do problema e oportunidade de pesquisa, (2) definição dos objetivos para uma solução do problema, (3) design e desenvolvimento do artefato, (4) demonstração da viabilidade do artefato, (5) avaliação do artefato e (6) comunicação e disseminação dos resultados.

Essas etapas são cíclicas, pois a avaliação do artefato pode levar a revisões e refinamentos do problema, dos objetivos e do próprio artefato, reiniciando o ciclo [26]. O DSR é amplamente utilizado em áreas como a tecnologia da informação, engenharia e ciência da computação para a criação e validação de artefatos de design práticos e úteis.

O objetivo da pesquisa em questão é explicativo pois a partir de regras das recomendações da AMIB/ABRAMEDE de alocação de recursos em

esgotamento durante a pandemia por COVID-19 publicado, pode-se comprovar que com dados simulados o protocolo funciona na maior parte dos casos, assim como utilizado tecnologia para uma listagem que auxilia o médico. Os procedimentos desta pesquisa são considerados experimentais, pois é definido um objeto de estudo, no caso os dados validados por médicos.

### 2.1.2 Método de trabalho

A pesquisa adotou o método *Design Science Research* (DSR), que é caracterizado por um ciclo de vida iterativo e incremental. Esse ciclo começa com a identificação e definição do problema a ser resolvido e termina com a validação do artefato produzido. Para garantir um processo eficiente e bem-estruturado, a pesquisa foi organizada em fases seguindo o ciclo de vida do método DSR, que incluem a identificação e definição do problema, planejamento, construção, avaliação e conclusão. Cada fase teve atividades específicas, realizadas de forma iterativa e incremental para refinar e aprimorar o artefato produzido durante todo o processo. A Figura 2.1 apresenta as fases do estudo de forma visual. A seguir, é fornecida uma breve explicação para cada uma dessas fases.

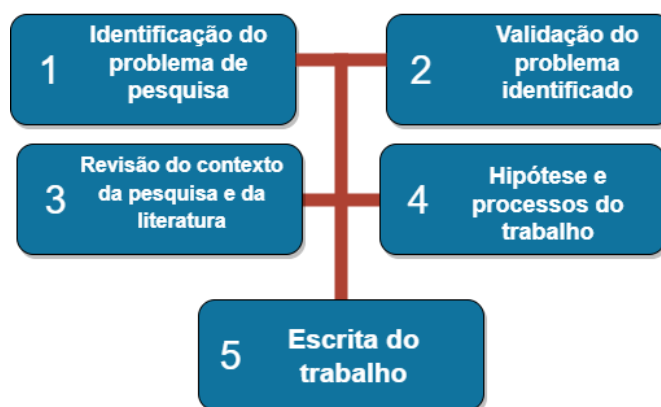


Figura 2.1: Fases do trabalho.

- i. Identificação do problema de pesquisa: Descreve o surgimento do problema de pesquisa, desde a proposta inicial até o afunilamento da pesquisa, com o objetivo de definir claramente o escopo e a relevância do estudo.
- ii. Validação do problema identificado: Busca verificar se o problema levantado é realmente relevante e se há uma demanda real para a sua

- solução. Realiza-se uma entrevista com especialista que fornece *insights* valiosos sobre as necessidades e demandas da área, contribuindo para a definição de um problema que realmente demande atenção e solução.
- iii. Revisão do contexto da pesquisa e da literatura: Essa fase é fundamental para identificar as questões relevantes e garantir a aplicabilidade relacionadas ao problema de pesquisa, bem como para contextualizar a abordagem adotada. A revisão da literatura permite a identificação de lacunas no conhecimento e de oportunidades de pesquisa, além de permitir o estabelecimento de um estado da arte e a comparação dos resultados obtidos com outros estudos na área.
  - iv. Hipótese e processos do trabalho: Nesta fase, é formulada uma hipótese com base na revisão bibliográfica e na análise crítica do problema em questão, que posteriormente é testada por meio de metodologias adequadas. Além da formulação da hipótese, é essencial descrever o processo completo de solução de forma clara e detalhada, de modo a garantir a compreensão adequada do mesmo pelos leitores e colaboradores. É importante destacar que a formulação de uma hipótese coerente e fundamentada é fundamental para o sucesso da pesquisa e para a obtenção de resultados confiáveis e relevantes.
  - v. Escrita do trabalho: Nesta etapa, é realizada a produção do texto científico, seguindo as normas e diretrizes estabelecidas pela comunidade científica, com clareza e objetividade na apresentação dos resultados obtidos. É importante ressaltar que a escrita deve ser contínua, com revisões e correções ao longo do processo de escrita, a fim de garantir a qualidade e a precisão do trabalho apresentado. Ademais, é imprescindível a realização de uma revisão final rigorosa para assegurar a consistência e a coerência do texto, e assim, contribuir para a validade e credibilidade da pesquisa científica.

## **Fase 1 - identificação do problema de pesquisa**

Durante o período deste mestrado, em 2021, em meio à pandemia de COVID-19 que ainda afetava o Brasil, identificou-se um problema crítico de escassez de leitos de UTI. Por meio de conversas com o médico Dr. José Salomão e revisão da literatura científica, buscou-se desenvolver uma solução para um dos principais desafios enfrentados pelos médicos intensivistas, a tomada de decisão sobre a alocação de pacientes nas UTIs. Essa decisão é fundamental para garantir o melhor atendimento aos pacientes e alocar adequadamente os recursos disponíveis. No entanto, a falta de uma metodologia

padronizada e eficiente pode levar a decisões subjetivas e falhas, prejudicando a qualidade do atendimento e a eficiência do sistema de saúde como um todo.

Diante desse problema, propomos o desenvolvimento de um sistema baseado em dados do paciente e de geolocalização, utilizando os protocolos *Acute Physiology and Chronic Health Evaluation* (APACHE) e Triagem de Processo Adaptável (ADAPT) por meio de um modelo de IA e equipamentos de Internet das Coisas para auxiliar médicos na decisão de internação na UTI. O sistema seria destinado à central de leitos do estado do Rio Grande do Sul, que gerencia as internações de UTI através do sistema GERINT.

Entretanto, após tentativas frustradas de reuniões com a equipe médica estadual, o escopo da atual pesquisa foi afunilado para um público específico: gestores de hospitais, médicos intensivistas e equipes de triagem de pacientes. O novo objetivo desta pesquisa foi aplicar o protocolo de triagem desenvolvido pela Associação de Medicina Intensiva Brasileira (AMIB), especializado em cenários com escassez de leitos de UTI. Com essa mudança, o problema atual do estudo foi definido e ajustado para atender às novas especificações.

## **Fase 2 - validação do problema identificado**

A validação do problema de pesquisa é uma etapa crucial para garantir que a solução proposta seja viável e útil para os profissionais da área de saúde. Nesse sentido, o processo de validação envolveu a realização de conversas com médicos intensivistas para compreender suas necessidades e desafios na tomada de decisão sobre quais pacientes devem ser encaminhados para a UTI.

Os resultados dessas conversas revelaram que os médicos enfrentam dificuldades em escolher quais pacientes devem ser encaminhados para a UTI, especialmente em situações de alta demanda e escassez de leitos. Essas decisões são complexas e devem levar em consideração diversos fatores, como a gravidade do paciente, o tipo de tratamento necessário, a disponibilidade de leitos e a probabilidade de sucesso do tratamento.

Para a validação do problema, foi realizada uma visita ao Hospital de Caridade de Santo Ângelo, no Rio Grande do Sul, para conversar com a médica responsável intensivista. Durante essa visita, foi possível confirmar o problema de pesquisa, bem como entender melhor a exaustão psicológica dos médicos em situações de calamidade pública.

A validação do problema de pesquisa garante que a solução proposta seja relevante e eficaz para os profissionais da área de saúde. A confirmação do

problema pelos médicos que vivenciaram o período de pandemia e a validação dos dados coletados reforçam a importância da pesquisa e da busca por soluções que possam auxiliar a equipe médica a tomar decisões mais precisas e ágeis em momentos de crise.

### **Fase 3 - revisão do contexto da pesquisa e da literatura**

Essa etapa foi conduzida com base em trabalhos previamente realizados no mundo acadêmico e também pela validação do depoimento dos médicos para garantir a veracidade.

Devido à dificuldade em obter acesso aos dados de pacientes reais, optou-se por gerar dados sintéticos, que foram validados por médicos especialistas. Essa abordagem permite uma análise em ambiente controlado do problema identificado, com a criação de cenários simulados e a aplicação de diferentes protocolos de triagem e avaliação clínica.

Dessa forma, a revisão do contexto da pesquisa foi uma etapa fundamental para a compreensão abrangente do problema de pesquisa e para a definição da abordagem metodológica que seria adotada para a criação da solução. Permitiu a validação da relevância e atualidade do problema em questão, reforçando a importância da pesquisa e seu potencial impacto na saúde pública.

Durante as etapas do trabalho, foram realizadas pesquisas bibliográficas a fim de fundamentar o conhecimento existente sobre o problema em questão e identificar as melhores soluções propostas na literatura científica. Foi realizada uma revisão sistemática e metódica da literatura científica disponível, a fim de fundamentar o conhecimento, comportamento e aplicabilidade do protocolo AMIB/ABRAMEDE, apresentando uma análise detalhada.

A revisão da literatura científica foi realizada utilizando palavras-chave específicas, como “Leito de UTI escasso”, “Protocolo Pandemia”, “Alocação de recursos em esgotamento”, dentre outras, e plataformas de busca, como ACM, SCOPUS, IEEE, ScienceDirect, Elsevier e PUBMED. Dessa forma, foram identificados trabalhos relevantes que abordam a problemática da pesquisa, bem como a utilização de IA em problemas de alocação de recursos em UTI.

Utilizou-se também uma tese de doutorado intitulada “Avaliação de um instrumento de auxílio à tomada de decisão para a priorização de vagas em unidades de terapia intensiva” de 2018, um modelo de protocolo: “Protocolo AMIB de alocação de recursos em esgotamento durante a pandemia por COVID-19” e um artigo: “*Intensive care unit (ICU) management based on big data*”.

A partir dessa revisão da literatura, identificou-se os principais algoritmos utilizados na alocação de recursos em UTI, bem como seus respectivos fundamentos. Dessa forma, possibilitou-se definir os requisitos necessários para a construção do artefato proposto, permitindo uma compreensão mais completa do seu funcionamento, contribuindo para a solução do problema identificado.

## Fase 4 - hipótese e processos do trabalho

Durante o processo de avaliação e validação do problema de pesquisa, bem como de revisões, surgiu a hipótese de que o uso de algoritmos de IA combinados com modelagem matemática pode resultar em um modelo computacional capaz de recomendar pacientes prioritários, baseado na segunda versão do protocolo AMIB/ABRAMEDE de 2020. Esse modelo pode oferecer sugestões de pacientes com maior transparência e imparcialidade, seguindo critérios pré-definidos e treinados, com o objetivo de auxiliar profissionais da saúde. Para o desenvolvimento dessa solução, foram definidos quatro processos, os quais seguem uma abordagem iterativa e incremental, visando garantir a qualidade e eficiência do artefato final.

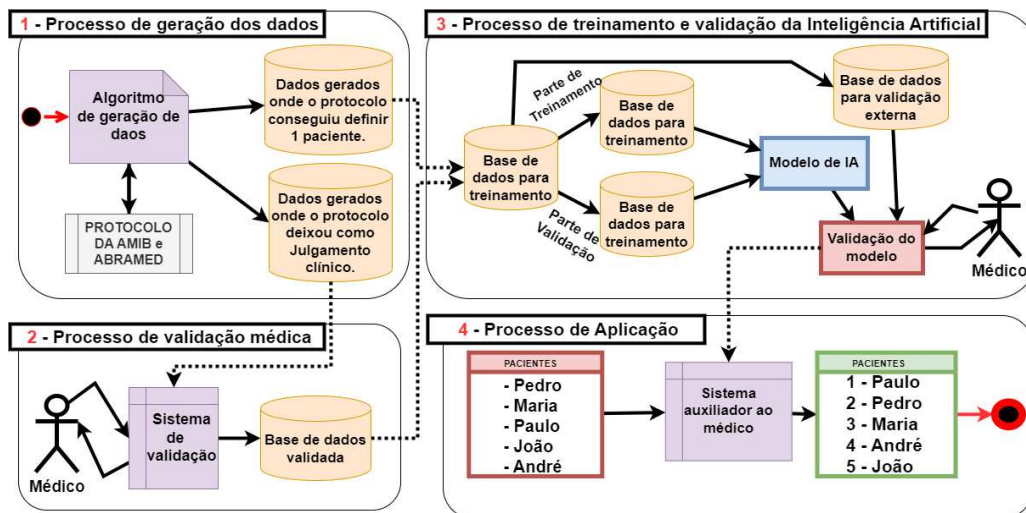


Figura 2.2: Processos de desenvolvimento.

Conforme ilustrado na Figura 2.2, serão apresentados a seguir os detalhes de cada processo, a fim de proporcionar uma compreensão mais abrangente e aprofundada das etapas envolvidas no desenvolvimento do estudo. O código fonte está disponível para consulta e uso no repositório do *Git Hub* [57].

1 - O processo de geração de dados envolve a criação de um algoritmo baseado na, recomendações da AMIB/ABRAMEDE de alocação de recursos em esgotamento durante a pandemia por COVID-19, para gerar registros de pacientes hipotéticos, representativos do problema a ser resolvido e em formato apropriado para processamento pela IA. De acordo com o protocolo AMIB, existem situações em que as pontuações são idênticas, requerendo decisões da equipe médica. Tais casos foram selecionados e preparados em um software (para uso no processo de validação médica), permitindo que médicos experientes validem e selecionem a melhor escolha, a fim de transmitir conhecimento ao modelo de IA. A outra parte dos dados, em que o protocolo permite a tomada de decisão, foi separada para ser utilizada no processo de treinamento e validação da IA.

2 - O processo de validação médica consiste na utilização de um software desenvolvido especificamente para receber os dados gerados pelo algoritmo com base no protocolo AMIB/ABRAMEDE, que foram marcados como casos que requerem julgamento clínico. Esse software foi disponibilizado para médicos voluntários, que realizaram a ordenação dos casos em um grupo de 10 pacientes. Os dados resultantes da ordenação foram então armazenados em uma base de dados dedicada e foram utilizados na fase subsequente do projeto, a fim de treinar e validar o modelo de IA. É importante destacar que a validação médica é fundamental para garantir a precisão e a eficácia do modelo, uma vez que os médicos podem identificar erros ou lacunas nos dados gerados pelo algoritmo.

3 - No processo de treinamento e validação da IA, a base de dados unificada, composta por dados validados no processo 2 e por dados gerados com decisões no processo 1, é dividida em 70% para treinamento e 30% para teste. O algoritmo *random forest* é empregado para o treinamento do modelo, permitindo que este reconheça padrões e tome decisões com base nos dados fornecidos. Durante o treinamento, são realizadas experimentações e monitoramento do desempenho, com ajustes efetuados conforme necessário. Após a validação da base de dados, o modelo de IA é exportado em um arquivo e é submetido a uma validação adicional com dados inéditos para avaliação de sua precisão de acerto. Nesta fase, alguns médicos também são convidados a realizar validações em amostras para validação.

4 - No processo de aplicação, o modelo de IA treinado é aplicado para resolver o problema em questão. Ao inserir uma lista de pacientes, o modelo pode ordená-los rapidamente com base em sua prioridade, auxiliando o médico na escolha dos pacientes mais urgentes. Esse processo é possível devido ao aprendizado do modelo sobre os padrões de dados relevantes e a capacidade de tomar decisões precisas com base nesses padrões. No entanto, é

importante ressaltar que a decisão final cabe ao médico responsável, sendo a IA uma ferramenta de suporte à tomada de decisão.

### **Fase 5 - escrita do trabalho**

A escrita do trabalho ocorreu simultaneamente ao processo de pesquisa, onde a cada etapa concluída, os resultados obtidos foram documentados. Após a conclusão de todas as etapas da pesquisa, a escrita do trabalho foi revisada e ajustada com base nos resultados obtidos. Cada passo da pesquisa foi descrito detalhadamente, desde a definição da pesquisa e das hipóteses até a validação do modelo de Inteligência Artificial e a aplicação proposta.

Para garantir a precisão e clareza das informações apresentadas no trabalho, foram utilizadas as melhores práticas de escrita científica, incluindo referências bibliográficas adequadas, linguagem técnica apropriada e uma estrutura coerente e organizada. A escrita do trabalho foi essencial para documentar todo o processo de pesquisa, bem como para comunicar os resultados e conclusões para a comunidade científica e outros interessados.

## **2.2 Resumo do capítulo**

Este capítulo apresentou a metodologia utilizada nesta pesquisa, discutiu sua importância e esclareceu o método de trabalho e seus processos. A combinação dessas duas abordagens permitiu a condução eficaz da pesquisa, garantindo a integridade dos resultados obtidos.



---

## Capítulo 3

# Referencial Teórico

---

*Think like a wise man,  
but communicate in plain language.*

*William B. Yeats, Irish dramatist & poet (1865-1939)*

**E**ste capítulo apresenta o referencial teórico da pesquisa, explorando conceitos fundamentais relevantes para o campo da saúde e tecnologia. São discutidos o fluxo do processo de admissão do paciente e o protocolo de alocação de recursos da AMIB/ABRAMEDE e apresentado seus critérios. São abordados também tópicos conceituais relacionados à inteligência artificial e suas sub-áreas.

### 3.1 Fluxo e protocolo de triagem de pacientes na UTI

A Unidade de Terapia Intensiva (UTI) é um setor crucial em hospitais, onde pacientes graves e críticos são tratados. O gerenciamento eficiente de recursos é fundamental para garantir que os pacientes recebam o atendimento adequado e a melhor chance de recuperação, sendo necessário um fluxo organizado de pacientes e um protocolo de triagem rigoroso para alocar recursos de forma eficiente. Nesta seção, será apresentado o fluxo de pacientes que entram na UTI e o protocolo de triagem de alocação de recursos, destacando sua importância.

### 3.1.1 Fluxo do processo da admissão do paciente

Quando um leito é liberado, o médico responsável enfrenta a difícil tarefa de decidir qual paciente será admitido, muitas vezes tendo que escolher entre aqueles com poucas chances de sobrevivência. Esse é um momento de grande incerteza e ansiedade para todos os envolvidos.

O fluxo de admissão de um paciente em uma UTI geralmente envolve vários passos e pode variar de acordo com a instituição de saúde em questão. A seguir, é descrito um fluxo típico de admissão de um paciente em uma UTI.

Tipicamente o paciente é encaminhado para a UTI por um médico ou outro profissional de saúde. Isso pode acontecer após uma internação hospitalar ou após um procedimento médico, como uma cirurgia. Antes de ser internado, o paciente passa por uma avaliação detalhada realizada pelo médico responsável, incluindo a verificação de sinais vitais e um exame físico completo. Alguns exames de sangue podem ser solicitados e, em casos específicos, é possível que sejam necessários exames de imagem, como radiografias ou tomografias.

Após aferições e triagem dos pacientes, o médico responsável pela UTI é confrontado com a complexa tarefa de selecionar quais pacientes devem ocupar as vagas disponíveis, em um contexto de alocação de recursos limitados durante pandemias. Além de avaliar a condição médica dos pacientes, fatores prognósticos, como a probabilidade de sobrevivência e a capacidade de resposta ao tratamento, também precisam ser considerados para garantir uma alocação eficiente e justa dos recursos disponíveis.

A alocação de leitos de UTI durante pandemias é um processo delicado e complexo que exige um equilíbrio cuidadoso entre a equidade e a eficiência na utilização dos recursos limitados. Para atingir esse equilíbrio, é necessário o envolvimento de uma equipe interdisciplinar de profissionais de saúde e gestores hospitalares, que trabalhem em conjunto para otimizar a utilização dos recursos disponíveis, por meio da aplicação de critérios objetivos e transparentes de triagem e alocação de pacientes.

Após o paciente estar alocado, o médico responsável pela UTI determina o tratamento adequado para o paciente, conectando a equipamentos de monitoramento, como monitores cardíacos ou de pressão arterial, para acompanhar sua condição de perto. Enquanto estiver na UTI, o paciente será acompanhado de perto por uma equipe de profissionais de saúde, incluindo médicos, enfermeiros e técnicos de enfermagem.

O paciente é avaliado regularmente pela equipe médica responsável para verificar seu progresso e fazer ajustes no tratamento, se necessário. Quando o paciente estiver estável o suficiente, ele poderá ser transferido para um leito de enfermaria ou para sua casa.

É importante lembrar que o fluxo de admissão de um paciente em uma UTI pode variar de acordo com a instituição de saúde e a condição específica do paciente. Quando possível, a equipe de saúde irá trabalhar em estreita colaboração com o paciente e seus cuidadores para garantir que todas as decisões de tratamento sejam tomadas de forma cuidadosa.

### 3.1.2 Protocolo de alocação de recursos AMIB/ABRAMEDE

A alocação justa e transparente de recursos em situações de emergência é crucial para garantir que pacientes com maior chance de sobrevivência recebam tratamento adequado. Em resposta à pandemia de COVID-19, a Associação de Medicina Intensiva Brasileira (AMIB) publicou um protocolo em 2020 que estabelece critérios claros e fundamentados para a alocação de recursos em situações de esgotamento [31]. O protocolo segue as leis brasileiras e é eticamente justificável, proporcionando uma base para decisões claras e transparentes em tempos de crise.

A escassez de leitos de UTI no Brasil é histórica, e a COVID-19 agravou ainda mais a situação, aumentando a demanda por esses recursos. A avaliação e seleção adequadas de pacientes são essenciais para reduzir o número de mortes causadas por doenças e garantir que os recursos disponíveis sejam utilizados de forma justa [45]. Ao adotar procedimentos claros e transparentes, como os estabelecidos no protocolo da AMIB, a distribuição adequada de recursos é favorecida, minimizando os prejuízos em situações de escassez. A adoção desses procedimentos é recomendada por organizações de saúde, pois pode beneficiar o maior número possível de pessoas e garantir que as decisões de alocação de recursos sejam justas e baseadas em critérios objetivos e bem definidos [15].

Durante momentos não pandêmicos, a disponibilidade de leitos de UTI em serviços públicos de saúde é baseada em dois fatores: a necessidade de terapias de suporte orgânico e a probabilidade de recuperação do paciente. Nesses cenários, as decisões relacionadas ao plano de cuidados dos pacientes são centradas no indivíduo, e podem ser discutidas com os pacientes, seus familiares e a equipe assistente, a fim de refletir as características clínicas e as preferências dos envolvidos. É importante notar que não é considerada uma infração ética ou legal não oferecer medidas de suporte orgânico a pacientes em fase terminal.

No entanto, em momentos pandêmicos, como o recente cenário do COVID-19, a demanda por leitos de UTI e ventiladores mecânicos tem excedido a oferta disponível, fazendo com que a atenção da equipe médica se volte para a redução do número de mortes na população como um todo. Nesses casos, a triagem de pacientes é uma etapa crucial para minimizar o problema, uma vez que sua utilização contribui para a redução do número de mortes em nível populacional. Em crises sanitárias, o processo de triagem se torna ainda mais importante para minimizar os danos, já que sua implementação pode contribuir para reduzir o número de mortes em nível populacional.

Diante da pandemia, a AMIB, propôs um protocolo que inclui um modelo de triagem baseado em um modelo de aceitabilidade social [4] e ajustando as leis do país, sendo uma intersecção dentre três áreas:

- i. Ético, legais e aceitabilidade social: modelo que protege de potenciais questionamentos jurídicos.
- ii. Aplicabilidade: modelo complexo para oferecer uma segurança preditiva e que seja aplicável na prática.
- iii. Área técnica: modelo que auxilia o profissional de saúde a tomar decisões complexas associadas a alocação de leitos de UTI e ventiladores, retirando uma grande carga de responsabilidade deixando o processo de decisão mais transparente.

O objetivo do protocolo em questão, é salvar o maior número possível de vidas, no menor tempo e com menos recursos. Busca proteger os profissionais de saúde de possíveis implicações morais e jurídicas decorrentes da tomada de decisão, garantindo a transparência do processo de seleção e responsabilidade perante a sociedade. Este protocolo é aplicável a todos os pacientes, sem distinção.

O protocolo lista as condições essenciais para o início da implementação das regras, sendo elas:

- i. o reconhecimento de estado de emergência em saúde pública;
- ii. o reconhecimento de que tenha havido esforços razoáveis em aumentar a oferta dos recursos em esgotamento;
- iii. a criação de comissões de triagem hospitalares pelos diretores técnicos;

- iv. o alinhamento da gestão do protocolo intra-hospitalar com o sistema de regulação de leitos, locais/regionais que facilite a disponibilidade de leitos entre unidades hospitalares;
- v. a monitoração regular da condição de esgotamento de recursos de forma a identificar a necessidade do início da aplicação do protocolo bem como as condições para seu encerramento e;
- vi. o anúncio público do início e encerramento da aplicação do protocolo.

Amplamente difundida entre os profissionais de saúde, principalmente na área de intensivistas, a primeira versão criada do protocolo de triagem de alocação de recursos em esgotamento durante a pandemia por COVID-19, foi debatida sob o aspecto das variáveis extraídas, com o objetivo de melhorar o protocolo e ser uma alternativa em momentos críticos. Realizou-se uma atualização que resultou em uma segunda versão, com considerações relevantes da Associação Brasileira de Medicina de Emergência (ABRAMEDE), da Sociedade Brasileira de Geriatria e Gerontologia (SBGG) e Academia Nacional de Cuidados Paliativos (ANCP), melhorando a forma de escrita, justificando as mudanças com embasamento científico, e realizando mudanças, destaca-se a alteração de uma variável, e de um critério de desempate.

A nova versão teve a modificação do critério que diz respeito a idade do paciente, que por questões ética, jurídica e discriminatória, entendeu-se que altera-la pelo ECOG, como medida de funcionalidade do paciente, será mais eficaz em uma análise sob a condição de saúde do paciente. Ainda nesse aspecto, outra informação alterada, é o último critério de desempate caso os anteriores não foram possíveis de ser aplicados, na primeira versão, era definido uma randomização dos pacientes na espera. Na versão atual, o último critério é um julgamento clínico da equipe de triagem.

### **Critérios do protocolo**

O modelo de triagem considera três critérios clínicos: a avaliação da gravidade da situação aguda do paciente, com base nos órgãos comprometidos, calculada pelo SOFA; a presença de comorbidades graves, indicando um período previsto de vida limitado; e a funcionalidade do paciente, avaliada pela escala ECOG. A AMIB desenvolveu uma tabela que apresenta visualmente os valores correspondentes a cada critério, conforme demonstrado na Tabela 3.1.

CRITÉRIOS	PONTUAÇÃO			
	1	2	3	4
<b>Passo 1</b> - Calcular e pontuar conforme estratificação ao lado.	≤ 8	9 - 11	12 - 14	≥ 14
<b>Passo 2</b> - Tem comorbidades graves, com expectativa de sobrevida ≤ que um ano?	—	—	Sim	—
<b>Passo 3</b> - Aplicar a ECOG e pontuar conforme a estratificação ao lado	0 - 1	2	3	4
<b>Passo 4</b> - Calcular a pontuação total dos critérios				
<b>Passo 5</b> - Alocar o leito de UTI ou VM ao paciente com menor pontuação total desde que não tenha havido empate.				
<b>Passo 6</b> - Em caso de empate utilizar os seguintes critérios hierarquicamente:				
<b>Passo 6a</b> - Menor escore SOFA total.				
<b>Passo 6b</b> - Julgamento clínico da equipe de triagem.				

**Tabela 3.1:** *Citérios do protocolo de alocação de recursos AMIB/ABRAMEDE. (Adaptado de Conselho Federal de Medicina (CFM) [13]).*

### **Critério 1 - Sequential Organ Failure Assessment**

O primeiro critério do protocolo AMIB, é calculado pelo Sequential Organ Failure Assessment (SOFA), ele avalia objetivamente o avanço de uma disfunção orgânica em pacientes que apresentam alguma infecção, bem como a mortalidade diante de um estado de saúde crítico de um indivíduo, avaliando seis sistemas diferentes a partir de diversos exames clínicos e laboratoriais, e prediz a mortalidade de pacientes com sepse (presença de disfunção orgânica que ameaça a vida, por conta de uma resposta exacerbada a uma infecção) a partir da pontuação [33].

A pontuação é atribuída aos seis sistemas orgânicos corpo humano, são eles: respiratório, cardiovascular, hepático, renal, neurológico e de coagulação, com base em sua avaliação e classificação de zero a quatro pontos. A pontuação deve ser calculada 24 horas após a admissão do paciente e, subsequentemente, a cada 48 horas, conforme justificado pelo termo "sequencial". Cada sistema representa um componente vital do corpo humano e sua avaliação contínua é importante para monitorar a condição do paciente.

O valor zero representa que o sistema está normal, e quatro representa que o sistema está em um alto grau de disfunção ou falência. A escala SOFA tem uma pontuação variável entre zero e vinte e quatro (0-24), e existe uma correlação entre a pontuação SOFA e a porcentagem de mortalidade do paciente. A Tabela 3.2 apresenta essa correlação:

SOFA	MORTALIDADE
0-6	$\leq 10\%$
7-9	15-20%
10-12	40-50%
13-14	50-60%
15	$\geq 80\%$
15-24	$\geq 90\%$

Tabela 3.2: Pontuação SOFA por taxa de mortalidade.

- i. **Sistema neurológico:** recebe avaliação a partir da escala de coma de Glasgow (ECG), é uma escala de ordem neurológica capaz de medir e avaliar o nível de consciência de uma pessoa que tenha sofrido um traumatismo craniano. Esse método é regularmente utilizado durante as primeiras 24 horas após o trauma, muito confiável para detectar o nível de consciência de uma pessoa após acidentes. A avaliação da escala é baseada em três parâmetros: abertura ocular, resposta motora e resposta verbal. Cada parâmetro recebe uma pontuação e o somatório final resulta no diagnóstico do paciente. A classificação que varia de 3 a 8 pontos é considerada grave, tendo a necessidade de intubação imediata, de 9 a 12 pontos é considerada moderada e a classificação do ECG de 13 a 15 é considerada leve. A pontuação relacionada ao SOFA ao somatório do ECG, é da seguinte forma: 1 ponto: se total acima de 13; 2 pontos: se total acima de 10; 3 pontos: se total acima de 6; 4 pontos: se total for menor que 6; caso total de ECG for 15, não pontua no score SOFA [62].

Pontos	Definição
0	15
1	Entre 13 a 14
2	Entre 10 a 12
3	Entre 6 a 9
4	Menor que 6

**Tabela 3.3:** Classificação do sistema neurológico.

- ii. **Sistema respiratório:** avaliado a partir de dados obtidos através de gasometria arterial, coletado através de um exame de sangue em uma artéria, sendo mensurado pela razão entre PaO<sub>2</sub>/FiO<sub>2</sub>, os gases presentes no sangue, como o oxigênio o gás carbônico [42]. A pontuação relacionada ao SOFA aos valores capturados no sistema respiratório, é da seguinte forma: 1 ponto: se PaO<sub>2</sub>/FiO<sub>2</sub> abaixo de 400; 2 pontos: se PaO<sub>2</sub>/FiO<sub>2</sub> abaixo de 300; 3 pontos: quando o paciente está em suporte ventilatório com a PaO<sub>2</sub>/FiO<sub>2</sub> abaixo de 200; 4 pontos: quando a relação tem resultado menor que 100mmHg com suporte ventilatório.

Pontos	Definição
0	PaO <sub>2</sub> > 400 em ar ambiente
1	SpO <sub>2</sub> > 92% com Cateter nasal O <sub>2</sub> até 2l/min
2	SpO <sub>2</sub> > 92% com Cateter nasal O <sub>2</sub> até 5l/min
3	SpO <sub>2</sub> > 92% com ventilação mecânica com FiO <sub>2</sub> até 40%
4	SpO <sub>2</sub> > 92% com ventilação mecânica com FiO <sub>2</sub> > 40%

**Tabela 3.4:** Classificação do sistema respiratório.

- iii. **Sistema cardiovascular:** o paciente que apresenta hipotensão e necessidade de droga vasoativa (DVA) é quem pontua nesta parte do score. É mensurado a partir da Pressão Arterial Média (PAM), medido por mmHg. A pontuação relacionada ao SOFA aos valores capturados no sistema cardiovascular, é da seguinte forma: 1 ponto: se PAM abaixo de 70 mmHg; 2 pontos: é considerado o uso de DVA, se administração de dopamina < 5 µg/kg/min ou dobutamina em qualquer dose; 3 pontos: se administração de dopamina entre 5,1 e 15 µg/kg/min, noradrenalina menor que 0,1 µg/kg/min ou adrenalina menor que 0,1; 4 pontos: em caso de aumento da dosagem das DVAs; caso a PAM for 70

mmHg, não pontua no score SOFA [46].

Pontos	Definição
0	PAM > 70mmHg Sem uso de vasopressor
1	PAM menor que 70mmHg e Sem uso de vasopressor
2	Em uso de Dopamina $\leq 5$ ou Dobutamina qualquer dose
3	Em uso de Dopamina > 5 ou Noradrenalina $\leq 0.1$
4	Em uso de Dopamina > 15 ou Noradrenalina > 0.1

**Tabela 3.5:** Classificação do sistema cardiovascular.

- iv. **Sistema coagulação:** avaliado a partir do exame de contagem de plaquetas/trombócitos no sangue (plaquetograma  $10^3/\mu\text{L}$ ). A pontuação relacionada ao SOFA aos valores capturados no sistema de coagulação, é da seguinte forma: 1 ponto: se contagem abaixo de 150; 2 pontos: se contagem abaixo de 100; 3 pontos: se contagem abaixo de 50; 4 pontos: se contagem abaixo de 20; caso a contagem for 150.000/mm, não pontua no score SOFA.

Pontos	Definição
0	Maior que 150
1	Entre 100 a 149
2	Entre 50 a 99
3	Entre 20 a 49
4	Menor que 20

**Tabela 3.6:** Classificação do sistema de coagulação.

- v. **Sistema hepático:** realizada com o exame de bilirrubinas totais para avaliar o funcionamento do fígado, é uma substância alaranjada produzida quando o fígado decompõe glóbulos vermelhos velhos [33]. A pontuação relacionada ao SOFA aos valores capturados no sistema hepático, é da seguinte forma: se o valor é menor que 1.2, não pontua no score SOFA; 1 ponto: se valor abaixo de 1.9; 2 pontos: se valor abaixo de 5.9; 3 pontos: se valor abaixo de 11.9; 4 pontos: se contagem maior de 12.
- vi. **Sistema renal:** para a avaliação é considerado dois parâmetros: Creatinina ( $\mu\text{mol/L}$ ) e débito urinário (mL/dia). A pontuação relacionada ao SOFA aos valores capturados no sistema renal, é da seguinte forma: 1

Pontos	Definição
0	Até 1.2
1	Entre 1.2 a 1.9
2	Entre 2 e 5.9
3	Entre 6 e 11.9
4	Maior que 12

**Tabela 3.7:** *Classificação do sistema hepático.*

ponto: se creatinina abaixo de 1.2; 2 pontos: se creatinina abaixo de 1.9; 3 pontos: se creatinina abaixo de 4.9 ou diurese abaixo de 500 mL/dia; 4 pontos: quando diurese abaixo de 200 mL/dia ou creatinina maior a 5.0; caso creatinina abaixo de 1.2, não pontua no score SOFA [33].

Pontos	Definição
0	Creatinina menor que 1.2
1	Creatinina entre 1.2 a 1.9
2	Creatinina entre 2.0 a 3.4
3	Creatinina entre 3.5 a 4.9 ou diurese entre 199 até 500ml/dia
4	Creatinina maior que 5.0 ou diurese menor que 200ml/dia

**Tabela 3.8:** *Classificação do sistema renal.*

## **Critério 2 - Supportive and Palliative Care Indicators Tool**

O segundo critério citado pela AMIB pode ser calculado por duas ferramentas: *Supportive and Palliative Care Indicators Tool* (SPICT-BR) ou o PIG-GSF, ambas para entender a expectativa de sobrevida do paciente, para esse artigo foi escolhido usar a ferramenta SPICT-BR, que é uma lista de indicadores gerais de deterioração clínica e de indicadores de gravidades de doenças específicas: doenças cardiovasculares, doenças renais, respiratórias, hepáticas, câncer, doenças neurológicas e demências e fragilidades [54].

Para que haja a indicação de cuidado paliativo do paciente necessita pontuar pelo menos 02 ou mais dos indicadores gerais e 01 indicador específico. O SPICT-BR foi traduzido e validado para utilização em pacientes brasileiros em abril de 2016. A pergunta surpresa, corresponde ao questionamento – “Você acredita que o seu paciente tem expectativa de vida menor do que

12 meses?”, direcionado à equipe de saúde, mas especificamente, ao médico. Se a resposta for “Sim”, conforme o protocolo AMIB, receberá 3 pontos no total, caso contrário não somará.

### **Critério 3 - *Eastern Cooperative Oncology Group***

O *Eastern Cooperative Oncology Group* (ECOG) é o terceiro critério do protocolo AMIB, é um instrumento validado e amplamente utilizado em oncologia e que busca quantificar a capacidade funcional física e capacidade de independência e autocuidado do paciente. A inferência é que quanto pior a escala de performance do paciente menor sua reserva fisiológica e piores os desfechos clínicos.

A coleta desta medida deve ser referente à performance que o paciente exiba nas duas a quatro semanas que antecederam a internação de maneira a excluir o fator confundidor da presença de doença aguda que possa ter se iniciado nas duas semanas imediatamente anteriores a internação hospitalar. Não é recomendado o uso deste critério, sem uma avaliação clínica individualizada, a pacientes portadores de deficiências físicas de longa data e que apresentem uma boa condição de adaptação [50].

## **3.2 Inteligência Artificial**

A inteligência artificial (IA) não tem somente uma definição, pode-se dizer de forma geral que é um subcampo da ciência da computação que tem como objetivo desenvolver soluções para um problema, simulando a capacidade humana de sugerir soluções e/ou tomar decisões para situações específicas [55]. Para os autores Charniak [11], a IA é a “faculdade mental” que aprende através de modelos computacionais. Para o inventor e futurista Ray Kurzweil [32] a IA é a arte de criar máquinas, para que possam executar funções que precisam de inteligência. A IA é baseada em técnicas de aprendizado de máquina, que permitem que os sistemas aprendam por meio de experiência e adaptem suas ações de acordo com o ambiente em que estão inseridos [5].

A inteligência artificial em suas subáreas possui várias capacidades, como:

- Reconhecer padrões.
- Reconhecer imagens.

- Compreender uma linguagem escrita ou falada.
- Fazer/perceber relações entre eventos que um humano não faria.
- Obedecer a algoritmos criados para fazer decisões como se fosse um especialista.
- Entender os conceitos passados e não somente processar dados.
- Adquirir “raciocínios” pela capacidade de melhorar o próprio conhecimento e integrar novas experiências com o objetivo de resolver problemas ou realizar tarefas específicas [35].

De acordo com a pesquisa do Professor Cozman Cozman [16], a inteligência artificial tem uma ampla gama de aplicações, incluindo representação de conhecimento e raciocínio, tomada de decisão, sistemas multiagentes, aprendizado de máquina, a interseção da inteligência artificial com a sociedade, processamento de linguagem natural, visão computacional e robótica.

A IA é uma ferramenta poderosa que pode ser utilizada para avaliar os resultados de experimentos biológicos, requerendo informações sobre o tipo de resultados esperados e as características dos organismos envolvidos, além de ser considerada uma ferramenta especializada na análise de experimentos biológicos, devido à sua capacidade de simular habilidades humanas [16]. O subcampo de estudo conhecido como *machine learning* é especialmente importante, pois permite a automatização da criação de modelos analíticos. Em resumo, a Inteligência Artificial é uma tecnologia que pode ser aplicada com sucesso na avaliação de experimentos biológicos e na criação de modelos analíticos [12].

## Machine learning

*Machine Learning* (ML) é um campo da ciência da computação que se dedica a pesquisa e desenvolvimento de sistemas capazes de aprender por meio de experiência e adaptar suas ações de acordo com o ambiente em que estão inseridos [5]. ML é uma subárea da inteligência artificial (IA) que visa criar sistemas capazes de realizar tarefas de forma autônoma, sem a necessidade de intervenção humana [55]. O objetivo do ML é criar sistemas capazes de aprender por meio de dados e de aplicar esses conhecimentos em tarefas específicas, como previsão de resultados, classificação de objetos identificando padrões como também tomar decisões com ou sem a intervenção humana [2].

O criador do conceito, Arthur Samuel define ML como: “Um campo de estudo que dá aos computadores a habilidade de aprender sem terem sido programados para tal” [3]. A ML requer um conjunto de características como entrada, e entrega uma saída com o critério de agrupamento ou classificação [58].

O reaproveitamento do aprendizado de máquina em aplicações de diversas áreas é comum atualmente, criando um modelo de *machine learning*. O modelo resulta em um algoritmo especialista, que analisa dados históricos de um determinado assunto e pode-se aplicar um algoritmo que fará a predição sobre valores ainda não vistos, ou seja, a partir de um histórico, pode-se criar um modelo para prever o que possivelmente irá acontecer [58]. Existem três tipos de aprendizado de máquina: aprendizado supervisionado, aprendizado não supervisionado e semi-supervisionado, detalhados a seguir.

### **Aprendizado Não Supervisionado**

Neste tipo de aprendizado, a máquina não tem interação humana e deve entender o que tem que fazer com os dados de entrada, mas não tem uma classificação de saída, tendo de agrupar os dados da maneira lógica conforme o algoritmo utilizado [10, 61].

A aprendizagem não supervisionada ocorre quando um algoritmo aprende com um exemplo, sem a supervisão de um humano [65]. Ele tende a classificar os dados por grupos que não são previstos mas que trazem um padrão que o algoritmo percebeu. É usado com dados que não possuem as variáveis target de saídas. Durante o treinamento ele não recebe os dados de saída, fazendo com que os algoritmos tenham que descobrir qual a melhor saída, normalmente classificando os dados [61]. O objetivo é estruturar os dados e encontrar um padrão neles conforme o modelo, ressaltando seu funcionamento em dados transacionais, com atributos semelhantes.

### **Aprendizado Semi-supervisionado**

Normalmente utilizada em grandes bases de dados, quando é preciso mesclar a classificação da máquina, juntamente com a interação humana, realizando a classificação, mesclando as duas categorias anteriores [10, 60, 61]. Portanto, o termo aprendizagem de máquina é usado quando um programa pode automaticamente encontrar padrões e relações em um conjunto de dados, sem que o programa possua qualquer conhecimento de saída dos dados [65].

## Aprendizado Supervisionado

A máquina é ensinada por um humano, inserindo os dados de entrada e de saída. Normalmente é a mais utilizada, porque o humano tem o controle da classificação dos resultados finais [60].

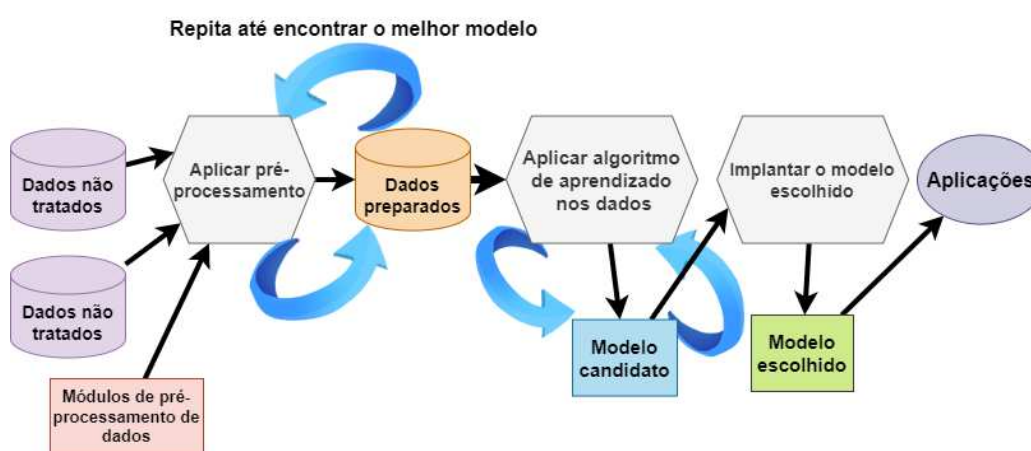
O aprendizado supervisionado é uma abordagem da inteligência artificial que se assemelha ao processo de aprendizado humano, no qual os indivíduos são treinados com exemplos rotulados e aprendem a prever os rótulos (também conhecidos como “*labels*”) em novas entradas de dados não rotulados. Esta abordagem pode ser comparada ao processo de ensino de uma criança de cerca de 4 anos, em que os pais fornecem exemplos rotulados e a criança aprende a associar os rótulos às entradas de dados. O aprendizado supervisionado é utilizado em várias áreas, como reconhecimento de padrões, classificação de objetos e previsão de resultados [8, 10].

Cada algoritmo de aprendizado procura por padrões nos dados de treinamento para que possam ser utilizados para prever a saída para novas entradas de dados. Cada algoritmo pode ser adequado para a procura de diferentes tipos de padrões. Depois de treinado com um conjunto de dados rotulados, o algoritmo é testado em um conjunto de dados não rotulados para avaliar sua precisão na previsão da saída. O aprendizado automático é comumente utilizado em sistemas que possuem dados históricos, pois pode ser utilizado para prever possíveis acontecimentos futuros [36].

A aprendizagem supervisionada se divide em duas subcategorias:

- **Classificação** - tem como objetivo atribuir rótulos a entradas de dados qualitativas ou categóricas. O algoritmo de classificação é treinado com um conjunto de dados rotulados e, em seguida, é utilizado para prever o rótulo para novas entradas de dados. A classificação é frequentemente utilizada quando as previsões são de natureza distinta, como por exemplo, *tweets* positivos e negativos ou classificação de gênero a partir de uma foto [36].
- **Regressão** - tem como objetivo prever valores numéricos específicos a partir de entradas de dados quantitativas. O algoritmo de regressão é treinado com um conjunto de dados rotulados e, em seguida, é utilizado para prever o valor numérico para novas entradas de dados. A regressão é frequentemente utilizada para prever variáveis quantitativas, como por exemplo, o peso, a altura, o preço de um produto ou o tamanho de um objeto [61].

Para aumentar a eficiência em aplicações de diversas áreas, é criado um modelo de *machine learning* baseado em algoritmos especialistas que analisam dados históricos de um determinado assunto. Este modelo é utilizado para aplicar um algoritmo que fará previsões sobre valores ainda não vistos, ou seja, a partir de um histórico de dados, é possível criar um modelo que preveja o que possivelmente irá acontecer no futuro. O objetivo do ML é criar sistemas capazes de aprender por meio de experiência e adaptar suas ações de acordo com o ambiente em que estão inseridos [10].



**Figura 3.1:** Fluxo do modelo de machine learning. (Adaptado de Chappell [10], Souza [61]).

A fase da coleta de dados trata do material que será usado durante o processo de aprendizado, (banco de dados, planilhas, texto). As informações devem ter uma quantidade grande e serem de qualidade alta, para que o modelo final tenha uma boa precisão de predição [10]. A fase seguinte é a preparação dos dados, um processo essencial para melhorar a qualidade dos dados de entrada, é feita a verificação das informações coletadas na fase anterior.

Essa fase demanda muito tempo para ser completada. Quanto maior a quantidade de dados e forma de estrutura, mais tempo será gasto pelo cientista de dados para finalizá-la. Chappell [10], destaca que 75% dos dados são para treinamentos, e 25% são para a fase de avaliação. São muitas técnicas utilizadas para melhorar a qualidade, dentre elas, entendimento da estrutura e dos dados, realização da limpeza, eliminação de dados desnecessários, recodificação, organização, tratamento de dados faltantes, afim de obter uma base de dados limpa para que o algoritmo faça a predição mais exata [61].

Na fase da escolha/treinamento do modelo, é aplicado o algoritmo escolhido nos dados já transformados na fase anterior. Existem uma infinidade de modelos de aprendizado de máquina disponíveis, mas cada um com suas características.

A etapa de treinamento é fundamental para preparar a máquina como também para aprimorar as suas habilidades de previsão. É nessa etapa que a máquina efetivamente aprende com seus erros e torna-se cada vez mais aperfeiçoada [10]. Após a máquina entender o que tem que fazer, com o modelo e os dados, é aplicada a fase de avaliação, que consiste na precisão do modelo, com as informações não utilizadas no treinamento, os 25% dos dados [10]. Essa fase é realmente uma avaliação para testar se a máquina aprendeu, ou somente memorizou os dados passados.

Aperfeiçoando o aprendizado da máquina, inicia-se a fase do aprimoramento dos parâmetros, que tem por objetivo melhorar a eficiência e conseqüentemente a qualidade do modelo. Essa etapa identifica valores de variáveis que afetam diretamente a acurácia do modelo e o tempo de treinamento necessário. Pode ser o ponto de reinício, utilizado para refazer todo o processo quando o algoritmo utilizado não é satisfatório [10, 61]. Por fim a fase de predição pode ser aplicada, pois a máquina está preparada para responder as tarefas que foram programadas [61].

### 3.2.1 Árvore de decisão

Árvore de decisão é um algoritmo de aprendizado automático que cria uma estrutura em forma de árvore para representar decisões e suas possíveis conseqüências [25] baseadas em atributos de um conjunto de dados. Cada nó da árvore representa uma decisão a ser tomada e cada ramo representa uma conseqüência possível dessa decisão [44].

O objetivo do algoritmo de árvore de decisão é construir um modelo de aprendizado supervisionado que permita realizar previsões sobre a saída para novos dados de entrada, baseado nas relações entre entradas e saídas aprendidas a partir de um conjunto de dados rotulados [5]. A árvore de decisão é um algoritmo de aprendizado supervisionado que permite realizar previsões devido à sua capacidade de visualizar a lógica subjacente à tomada de decisão bem como sua facilidade de interpretação e implementação [25, 51]. Além disso, as árvores de decisão são fáceis de interpretar e implementar, tornando-as uma escolha popular para muitos problemas de aprendizado de máquina [28].

O processo de criação de uma árvore de decisão começa com a escolha do atributo que melhor divide o conjunto de dados em subconjuntos mais homogêneos [56]. Em seguida, o processo é repetido para cada subconjunto até que todos os nós da árvore sejam folhas, ou seja, representem uma classe de saída. As folhas da árvore de decisão são utilizadas para realizar a previsão para novas entradas de dados, baseando-se nas decisões tomadas ao longo da árvore [63].

No entanto, pode ter dificuldades em lidar com dados muito ruidosos ou com atributos altamente correlacionados, o que pode resultar em uma árvore de decisão com muitos nós e baixa precisão. Para melhorar a robustez da árvore de decisão, é comum utilizar técnicas de *ensemble learning*, como o algoritmo de *random forest*, que combina várias árvores de decisão para obter uma previsão mais precisa.

### 3.2.2 Floresta aleatória

A Floresta Aleatória (*random forest* em inglês) é um algoritmo de aprendizado de máquina que combina múltiplas árvores de decisão para construir um modelo preditivo robusto. Foi introduzido por Leo Breiman em 2001 como uma extensão do método de árvore de decisão [17]. O algoritmo pertence ao conjunto de técnicas de *ensemble learning*, que utiliza a combinação de múltiplos modelos para melhorar a precisão das previsões.

### Ensemble learning

*Ensemble learning* é uma técnica de aprendizado de máquina que visa combinar diferentes modelos de aprendizado de máquina para obter melhores resultados do que os obtidos por um único modelo. A ideia básica do *ensemble learning* é que diferentes modelos podem aprender diferentes características do conjunto de dados e, combinando suas previsões, é possível obter uma melhor performance do que a obtida por cada modelo individualmente [20].

O *ensemble learning* é amplamente utilizado em várias áreas, incluindo classificação, regressão, detecção de anomalias, e muito mais. Existem diferentes tipos de técnicas, incluindo *bagging*, *boosting*, *random forests*. Cada técnica utiliza uma abordagem diferente para combinar os modelos individuais [20]. O *bagging* (ou *bootstrapped aggregating*) consiste em treinar vários modelos independentes a partir de amostras aleatórias do conjunto de dados. A previsão final é obtida ao agregar as previsões individuais de cada modelo.

Ele é amplamente utilizado para reduzir a variação dos resultados e aumentar a estabilidade do modelo [66]. O *boosting* consiste em treinar modelos sucessivos, onde cada modelo é treinado para corrigir os erros cometidos pelos modelos anteriores. É utilizado para melhorar a performance do modelo e aumentar a capacidade de aprendizado do conjunto de dados [22]. O *random forest* consiste em construir várias árvores de decisão independentes e combinar suas previsões. A previsão final é obtida ao agregar as previsões das árvores individuais, melhorando a performance do modelo, reduzindo a variação dos resultados, e aumentando a estabilidade do modelo [6]. O *ensemble learning* tem mostrado ser eficaz em muitas aplicações, incluindo classificação, regressão e detecção de anomalias. Isso se deve à capacidade de combinar diferentes perspectivas sobre os dados, o que pode resultar em uma performance superior em comparação a usar apenas um modelo isolado. É menos suscetível a *overfitting*, uma vez que as previsões individuais são combinadas para produzir uma previsão final. Apesar de algumas limitações, é amplamente utilizado na indústria e na pesquisa [25].

## Random forest

É um algoritmo de aprendizado de máquina que combina várias árvores de decisão para construir um modelo preditivo preciso e robusto. Cada árvore é construída a partir de um subconjunto aleatório dos dados de treinamento e um subconjunto aleatório das características, e utiliza uma técnica conhecida como *bootstrap* agregação para melhorar a estabilidade e a precisão do modelo. Durante o processo de treinamento, as árvores são construídas para minimizar o erro de classificação ou a soma dos erros quadrados. A previsão final do *random forest* é obtida pela agregação das previsões de cada árvore, utilizando a votação majoritária para a classificação e a média para a regressão [34].

Durante o processo de treinamento, cada árvore é construída usando o método de árvore de decisão padrão, dividindo os dados em subconjuntos com base nos valores das características, e escolhendo a divisão que maximiza uma determinada medida de impureza, como o índice Gini ou a entropia [6].

Para realizar uma previsão para uma nova amostra, o *random forest* utiliza a votação majoritária das previsões de cada árvore. No caso da classificação, a previsão final é a classe mais votada pelas árvores, enquanto que, na regressão, a previsão final é a média das previsões de cada árvore [6].

A agregação das previsões de cada árvore é realizada com base na importância das características. A importância de cada característica é medida

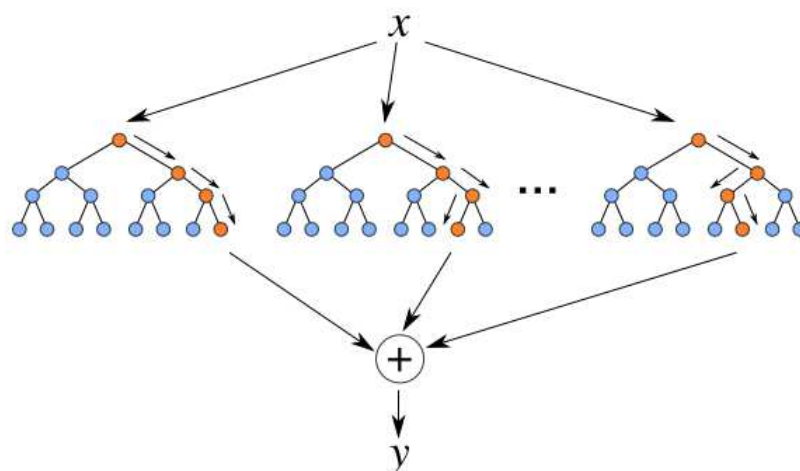


Figura 3.2: Representação de um *random forest* [1].

pela redução média da medida de impureza ao longo de todas as árvores, quando essa característica é usada para fazer uma divisão. Quanto maior a redução da impureza, maior a importância da característica [6].

Sendo um algoritmo eficiente, o *random forest* é capaz de selecionar automaticamente os atributos mais importantes para a previsão, o que o torna útil para tarefas de seleção de atributos. Na classificação, o algoritmo é usado para categorizar objetos em diferentes classes, como classificar e-mails como spam ou não spam [17]. Na regressão, é usado para prever variáveis contínuas, como o preço de uma ação ou a temperatura futura. Ela também é utilizada para determinar a importância de diferentes recursos em uma tarefa de previsão e também pode ser usado para selecionar automaticamente as características mais importantes em um conjunto de dados [23].

Ele combina as vantagens das árvores de decisão, como interpretabilidade e flexibilidade, com a agregação de previsões para melhorar a precisão e reduzir o *overfitting*. O algoritmo é especialmente útil em problemas com muitas características e uma grande quantidade de dados, pois é capaz de lidar com a complexidade e a dimensionalidade dos dados. Embora o algoritmo seja relativamente fácil de implementar e rápido de executar, há algumas desvantagens:

- i. Pode requerer consideráveis recursos computacionais quando os conjuntos de dados são muito grandes.
- ii. Pode ser menos eficaz em conjuntos de dados altamente desequilibrados.

- iii. Quando muitas árvores de decisão são usadas, a interpretação dos resultados pode ser complexa.

### **3.3 Resumo do capítulo**

Neste capítulo, foram introduzidos conceitos fundamentais da inteligência artificial, incluindo técnicas de treinamento e algoritmos. Foi apresentado o protocolo de triagem utilizado, explicando os critérios estabelecidos para garantir sua aplicabilidade, conformidade com a legislação e transparência para todos os envolvidos, juntamente com uma explicação detalhada do fluxo do paciente na entrada da UTI.

---

# Capítulo 4

## Modelo Computacional Proposto

---

*Knowing is not enough; we must apply.  
Willing is not enough; we must do.*

*Johann W. von Goethe, German Novelist (1749-1832)*

**N**este capítulo é apresentado de forma detalhada como a geração das bases de dados foram criadas, regras para a geração dos dados, geração de dados que o protocolo define julgamento clínico e a geração de uma massa de dados gerenciáveis. Também é apresentado o ambiente e suas bibliotecas utilizadas para a análise, e por fim o sistema de validação dos dados seletos que precisam de validação médica.

### 4.1 Visão geral

Em uma visão abrangente, pode-se considerar que o trabalho está dividido em quatro processos para a geração de dados e treinamento de um modelo de IA para auxiliar na alocação de recursos durante a pandemia da COVID-19. No primeiro processo, um algoritmo baseado no protocolo AMIB/ABRAMED é criado para gerar registros de pacientes hipotéticos. No segundo processo, médicos experientes validam os dados que requerem julgamento clínico. No terceiro processo, a base de dados é utilizada para treinar um modelo de IA, com monitoramento e ajustes conforme necessário. No último processo, o modelo treinado é aplicado para ordenar rapidamente uma lista de pacientes, mas a decisão final sempre cabe ao médico responsável. A validação médica é fundamental para garantir a precisão e a eficácia do modelo, uma vez que os médicos podem identificar erros ou lacunas nos dados gerados pelo algoritmo.

## 4.2 Geração dos datasets

Nesta seção, discutimos as fases de geração dos dados, que são primordiais para o desenvolvimento do modelo computacional proposto. A validação dos dados gerados matematicamente foi conduzida por profissionais médicos, cujo conhecimento especializado permitiu a identificação de informações prioritárias que poderiam ser relevantes para o modelo, mas que o protocolo AMIB/ABRAMEDE não seria capaz de discernir. Tal validação é fundamental para garantir a abrangência do treinamento dos dados utilizados como entrada para o modelo de inteligência artificial, permitindo que este seja treinado e ajustado adequadamente para tomar decisões precisas e confiáveis.

No início deste estudo, o objetivo da geração dos dados consistiu na realização de uma auditoria médica com o propósito de validar uma seleta base de dados em conformidade com o protocolo AMIB/ABRAMEDE a fim de garantir sua precisão no treinamento. Contudo, essa validação enfrentou algumas dificuldades tais como testemunho de alguns médicos, a pequena quantidade de informações disponíveis para cada paciente faltando o contexto em das informações e situação do paciente. Variáveis importantes, tais como idade, tempo de internação hospitalar, doença primária e tempo de espera por leito, foram identificadas como fatores cruciais para a avaliação adequada dos dados.

Diante das limitações encontradas durante a validação do protocolo AMIB/ABRAMEDE em algumas das validações realizadas, o processo de geração de dados foi dividido em fases distintas. Cada fase é cuidadosamente projetada para garantir que todas as variáveis relevantes sejam consideradas e que o processo seja executado com a máxima precisão possível. Além disso, cada fase é submetida a uma análise rigorosa para garantir que as informações geradas sejam confiáveis e precisas, e que os dados resultantes possam ser utilizados para fins clínicos relevantes.

**Fase 1 - Geração dos dados que o protocolo consegue definir o prioritário:** nessa fase o algoritmo foi gerado utilizando as variáveis e regras definidas pelo protocolo AMIB/ABRAMEDE, combinadas com todas as possíveis comparações em pares de pacientes. Como resultado, foram geradas 77.500 registros, sendo as possibilidades de pacientes restritas às variáveis informadas.

**Fase 2 - Geração de dados que o protocolo define julgamento clínico:** nessa fase gerou-se um banco de dados com 100 grupos de 10 registros

de pacientes, para a validação de profissionais médicos, com o objetivo de obter uma avaliação da ordem de prioridade indicada pelos especialistas e, posteriormente, aplicar esses dados no treinamento da inteligência artificial.

Fase 3 - **Geração da base de dados de comparação por pares**: considerando a limitação de máxima possibilidades de registros de pacientes na etapa anterior, foi aplicado o método de comparação por pares utilizando técnica binária, permitindo comparar todas as possibilidades dentro da base de dados. Como resultado, foi gerada uma base com mais de 6 bilhões de registros, contemplando todas as possibilidades de combinações em pares entre as variáveis dos pacientes.

Fase 4 - **Geração de uma massa de dados gerenciável**: com uma base grande, a dificuldade estrutural de treinamento, estava aumentando o custo computacional, portanto foi realizado a geração de nova base de dados com amostras de todas as escolhas geradas pelo protocolo AMIB/ABRAMEDE, incluindo explicitamente os dados de seleção dos médicos. Aproveitou-se para gerar dois arquivos com pacientes diferentes em cada arquivo. Com os arquivos será realizada a comparação e análise dos resultados obtidos pelos algoritmos com as escolhas realizadas pelos profissionais médicos, proporcionando uma avaliação mais completa e precisa da performance do modelo de inteligência artificial.

## Regras para a geração dos dados

As regras para a geração dos dados aplicadas no algoritmo criado, consistiram na definição de oito variáveis clínicas, bem como a aplicação de três critérios para alocação de recursos em esgotamento durante a pandemia por COVID-19, resultando em um modelo matemático que permitiu a geração de dados sintéticos para validação do algoritmo. A seguir serão denotados os critérios de  $\alpha_1$  até  $\alpha_8$ :

- $\alpha_1 \in \{0, 1, 2, 3, 4\}$ , sendo o critério neurológico
- $\alpha_2 \in \{0, 1, 2, 3, 4\}$ , sendo o critério respiratório
- $\alpha_3 \in \{0, 1, 2, 3, 4\}$ , sendo o critério cardiovascular
- $\alpha_4 \in \{0, 1, 2, 3, 4\}$ , sendo o critério coagulatório
- $\alpha_5 \in \{0, 1, 2, 3, 4\}$ , sendo o critério hepático
- $\alpha_6 \in \{0, 1, 2, 3, 4\}$ , sendo o critério renal
- $\alpha_7 \in \{0, 3\}$ , sendo o critério de comorbidades (SPICT)
- $\alpha_8 \in \{0, 1, 2, 3, 4\}$ , sendo o critério da escala de status de desempenho

O primeiro critério que compõe a pontuação total do protocolo, é o SOFA, demonstrado nas variáveis de  $a_1$  até  $a_6$ , que conforme os valores aferidos do paciente são atrelados a pontos para cada variável do SOFA, resultando em um somatório SOFA final, quantificando entre zero e quatro pontos.

Em seguida o modelo do protocolo AMIB/ABRAMEDE levanta o critério de comorbidades graves do paciente, o qual deve ser analisado a situação de cuidados paliativos juntamente com variáveis definidas na ferramenta SPICT, se caso a pergunta surpresa "Você acredita que o seu paciente tem expectativa de vida menor do que 12 meses?" for respondida como "Sim" será pontuado como três pontos no total do calculo AMIB/ABRAMEDE.

Para compor o último critério do protocolo AMIB/ABRAMEDE, calcula-se a escala de status de desempenho utilizando o ECOG, que busca quantificar a capacidade funcional física e capacidade de independência e auto-cuidado do paciente, quantificando entre zero e quatro pontos.

Para calcular a pontuação total ( $t$ ) dos três critérios de avaliação,  $C_1$ : Critério 1 - gravidade da situação aguda (SOFA);  $C_2$ : Critério 2 - comorbidades graves (SPICT-BR);  $C_3$ : Critério 3 - escala funcional (ECOG); formula-se a seguinte expressão matemática:

$$s = \sum_{n=1}^6 a_n, \text{ sendo o SOFA}$$

$$C_1 = \begin{cases} 1, \text{ se } & s \leq 8 \\ 2, \text{ se } 8 < & s \leq 11 \\ 3, \text{ se } 11 < & s \leq 14 \\ 4, \text{ se } 14 < & s \end{cases}$$

$$C_2 = \begin{cases} 0, \text{ se } & a_7 \neq \text{"Sim"} \\ 3, \text{ se } & a_7 = \text{"Sim"} \end{cases} \quad (4.1)$$

$$C_3 = \begin{cases} 1, \text{ se } & a_8 = 1 \\ 2, \text{ se } & a_8 = 2 \\ 3, \text{ se } & a_8 = 3 \\ 4, \text{ se } & a_8 = 4 \end{cases}$$

$$t = C_1 + C_2 + C_3$$

Com o objetivo de garantir a confiabilidade dos dados produzidos, foram implementadas normas rigorosas que evitam qualquer incongruência entre as informações coletadas e a realidade clínica. Estas normas incluem R1, R2 e R3, que foram estabelecidas com base no protocolo. Adicionalmente, foram incorporadas as regras R4, R5 e R6, fundamentadas em critérios clínicos avaliados por profissionais médicos.

$$\left\{ \begin{array}{l} R_1 : a_1, a_2, a_3, a_4, a_5, a_6 \in \{0, 1, 2, 3, 4\} \\ R_2 : a_7 \in \{0, 3\} \\ R_3 : a_8 \in \{0, 1, 2, 3, 4\} \\ R_4 : a_1 \geq 3 \implies a_2 \geq 3 \\ R_5 : a_2 = 1 \implies a_3 = 1 \\ R_6 : a_1 \geq 3 \implies a_8 = 0 \end{array} \right. \quad (4.2)$$

- A regra R1 estabelece que as variáveis  $a_1$  até  $a_6$  devem assumir valores entre zero e quatro, indicando que a avaliação SOFA deve ser levadas em consideração na avaliação clínica do paciente.
- A regra R2 indica que a variável  $a_7$ , que representa as comorbidades do paciente, deve assumir apenas os valores zero ou três, o que sugere que apenas comorbidades graves devem ser consideradas na avaliação do paciente.
- A regra R3 determina que a variável  $a_8$ , que indica o status de desempenho do paciente, deve assumir valores entre zero e quatro. Isso permite a avaliação da capacidade funcional do paciente, que é útil na tomada de decisão clínica.
- A regra R4 estabelece uma relação entre as variáveis  $a_1$  e  $a_2$ , indicando que, se o critério neurológico ( $a_1$ ) for igual ou maior que três, então o critério respiratório ( $a_2$ ) também deve ser igual ou maior que três. Isso sugere que a disfunção neurológica pode estar associada a problemas respiratórios.
- A regra R5 estabelece uma relação entre as variáveis  $a_2$  e  $a_3$ , indicando que, se o critério respiratório ( $a_2$ ) for igual a um, então o critério cardiovascular ( $a_3$ ) também deve ser igual a um. Isso sugere que a disfunção respiratória pode estar associada a problemas cardiovasculares.

- A regra R6 estabelece uma relação entre as variáveis  $\alpha_1$  e  $\alpha_8$ , indicando que, se o critério neurológico ( $\alpha_1$ ) for igual ou maior que três, então o status de desempenho ( $\alpha_8$ ) deve ser igual a zero. Isso sugere que a disfunção neurológica pode estar associada a uma redução na capacidade funcional do paciente.

## Geração de dados que o protocolo define Julgamento Clínico

O julgamento clínico, segundo o protocolo AMIB/ABRAMEDE, ocorre quando há empate tanto no escore SOFA quanto no escore total do protocolo. Nesse caso, o protocolo não é capaz de realizar uma escolha clara e precisa, evidenciando a necessidade de intervenção humana para a tomada de decisão. Diante desta regra, foi tentada a extração das decisões médicas a partir de uma base de dados contendo apenas os casos em que o protocolo não conseguiu discriminar entre as opções de decisão, a fim de analisar as escolhas dos profissionais em situações de empate.

A partir da análise das variáveis de acordo com os critérios estabelecidos pelo protocolo AMIB/ABRAMEDE, é possível criar a classe “Paciente”, que incorpora as oito variáveis descritas no modelo matemático. Essa classe permite a geração de dados sintéticos que podem ser usados para validar o modelo proposto, uma vez que simula as características dos pacientes reais de maneira controlada e precisa.

A criação do algoritmo desenvolvido, compreende as variáveis  $\alpha_1$  até  $\alpha_8$  e os critérios  $C_1$ ,  $C_2$  e  $C_3$  resulta em saídas onde é escolhido um paciente sem ocorrer empate, ou é realizado um empate da soma total, porém o critério de desempate SOFA resolve.

Com o objetivo de avaliar a eficácia do presente protocolo, optou-se por criar situações clínicas nas quais o protocolo seja incapaz de oferecer uma distinção significativa em relação à priorização de um paciente em detrimento de outro, especialmente quando o protocolo sugere a utilização de julgamento clínico, deixando a responsabilidade integral para os profissionais da saúde. É possível construir uma função  $\alpha \in [0, 1]$  que seja capaz de expressar o nível de confiança com que o protocolo determina a prioridade, de acordo com critérios estabelecidos. Essa função pode ser utilizada como ferramenta para avaliar e comparar diferentes protocolos e suas respectivas efetividades em relação à definição de prioridades. Notadamente, é possível formular uma função  $\alpha \in [0, 1]$  que informe o grau de certeza com o qual o protocolo determina a prioridade:

$$\alpha = \begin{cases} \frac{t_{p_1} - t_{p_2}}{\max_t - \min_t}, & \text{se } t_{p_1} - t_{p_2} \neq 0 \\ \frac{a_{7p_1} - a_{7p_2}}{\max_{a_7} - \min_{a_7}}, & \text{se } a_{7p_1} - a_{7p_2} \neq 0 \\ \frac{a_{8p_1} - a_{8p_2}}{\max_{a_8} - \min_{a_8}}, & \text{se } a_{8p_1} - a_{8p_2} \neq 0 \\ \frac{s_{p_1} - s_{p_2}}{\max_s - \min_s}, & \text{se } s_{p_1} - s_{p_2} \neq 0 \\ 0, & \text{se o protocolo julgamento clínico da equipe de triagem} \end{cases} \quad (4.3)$$

Mínimizar  $\alpha$  será trivial, considerando o caso de dois pacientes com todos os critérios idênticos, mas esse caso foge do que é desejável, pois não são cabíveis quaisquer outros argumentos quanto a priorização. Para tanto utiliza-se a noção de inércia ( $I$ ) como medida de homogeneidade, de forma que valores menores que  $I$  implicam em conjuntos de pacientes mais homogêneos. A inércia pode ser calculada como o quadrado da distância de cada membro do conjunto para o centro de gravidade do conjunto.

$$I = \sum_{p \in P} d(p, c)^2 \quad (4.4)$$

Considerando  $P$  o conjunto de pacientes,  $c$  o centro de massa desse conjunto e  $d$  a distância euclidiana, é possível a construção de um conjunto de dados que maximize a obtenção de quais são os critérios de decisão de um julgamento clínico, presumindo que utilize o protocolo AMIB/ABRAMEDE. Para tanto, qualquer que seja o conjunto de pacientes ordenado, esse deve maximizar a inercia e minimizar  $\alpha$ .

Considerando uma função de qualidade:

$$\kappa = \frac{I}{\alpha} \quad (4.5)$$

É construído um novo paciente  $P$  escolhendo um ponto nesse espaço de oito dimensões, nesse processo são feitas validações de  $a_1$  até  $a_8$  de forma que pertence aos conjuntos definidos, ou seja, as variáveis extraídas conforme o protocolo AMIB/ABRAMEDE.

No mesmo algoritmo é construído um conjunto de dez pacientes validados pelas regras citadas na Equação 4.2, para o conjunto é calculada a função  $\kappa$  assim como os dez subconjuntos de oito elementos.

No subconjunto com maior  $\kappa$  será complementado com um novo paciente gerado e validado se o  $\kappa$  ainda é maior  $\kappa$  do conjunto original. Desta forma, é possível gerar o paciente com melhor  $\kappa$ , para que os dados possam ser mais homogêneos possíveis, dentro de um conjunto, ou seja, dentro de um conjunto de dez pacientes, apresenta valores de dimensões diferentes mas com semelhança de estrutura no resultado, sendo  $\alpha = 0$ , obrigando os pacientes passarem por julgamento clínico conforme o AMIB/ABRAMEDE.

O registro hipotético validado na função com maior valor é utilizado para a persistência da informação, o processo fica em um laço de repetição definido por melhor desempenho computacional, por dez mil vezes, valor esse pré programado no código, dessa forma o algoritmo fornece um conjunto de paciente homogêneos e em que o protocolo AMIB/ABRAMEDE sugere que seja utilizado um julgamento clínico pelos profissionais.

Com base em uma amostra de mil registros de pacientes hipotéticos, foi observada uma homogeneidade na soma do SOFA, refletida na porcentagem de mortalidade desses pacientes. A Figura 4.1 demonstra que há semelhanças entre os conjuntos de pacientes, mesmo quando apresentam valores distintos.

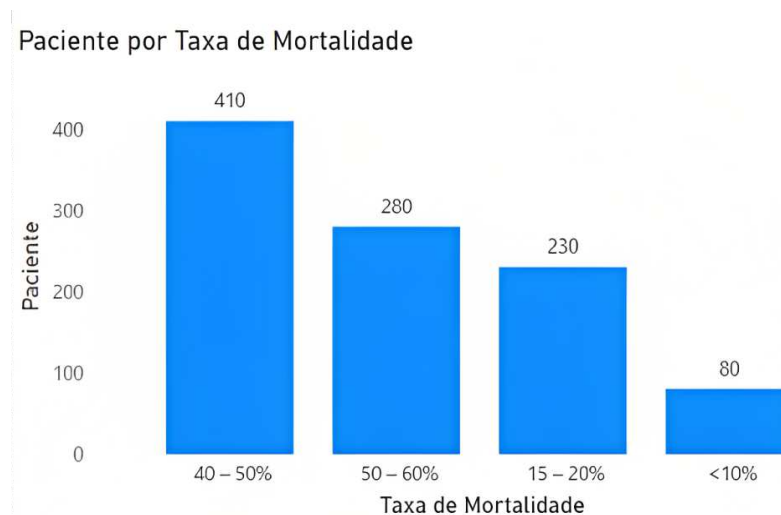


Figura 4.1: Paciente por taxa de mortalidade.

### Geração de dados com definição de prioridade pelo protocolo

Da mesma forma que a geração de dados anterior, utiliza-se a criação de uma classe intitulada "Paciente", a qual incorpora as oito variáveis previamente descritas no modelo matemático com base no protocolo

AMIB/ABRAMEDE. Essa abordagem permite a geração de dados sintéticos que serão a base de ensinamento para um modelo de inteligência artificial realizar o treinamento.

As regras da geração dos dados seguem as mesmas, realizando somente a modificação de que o produto deste algoritmo é a comparação em par de pacientes, que protocolo AMIB/ABRAMEDE classificou com o número 1 (um), se for o primeiro paciente da dupla o mais prioritário, com o número 2 (dois) se for o segundo paciente e com o número 0 (zero) se não consegue definir um mais prioritário.

Durante a geração da massa, foi possível identificar uma barreira na quantidade de registros, isso devido ao algoritmo não encontrar mais possíveis duplas para a geração. Habilmente com a aplicação das regras do protocolo AMIB/ABRAMEDE e um algoritmo em Python que processou a massa de dados, foi possível encontrar o número de 15 mil e 500 possibilidades de combinações possíveis.

Neste algoritmo está garantindo a transitividade, uma propriedade importante em muitos ramos da matemática e da lógica, pois permite estabelecer relações e inferências lógicas a partir de outras relações já conhecidas. Exemplificando na matemática, a relação “ser maior que” é transitiva. Se o paciente  $\chi$  é mais prioritário que o paciente  $\phi$  e  $\phi$  é mais prioritário que o paciente  $\psi$ , então conclui-se que  $\phi$  é mais prioritário que  $\psi$  pois essa relação é transitiva, porque ela “transita” entre os três elementos ( $\chi$ ,  $\phi$ , e  $\psi$ ).

Também é garantido no algoritmo gerado a propriedade anti-simetria, a qual é importante na lógica pois permite estabelecer propriedades importantes sobre relações binárias entre os elementos. Se uma relação  $R$  é anti-simétrica e reflexiva, então ela é uma ordem parcial. Uma ordem parcial é uma relação binária que é reflexiva, transitiva e anti-simétrica. As ordens parciais são importantes em muitos ramos da matemática e da ciência da computação, como na teoria dos conjuntos, na teoria das relações e na teoria dos algoritmos. Aplicado ao modelo em questão, se o algoritmo diz que o paciente  $\chi$  é mais prioritário que paciente  $\phi$ , o algoritmo deve confirmar quando o paciente  $\phi$  for registrado que é menos prioritário que o paciente  $\chi$ .

Dos 28 registros validados por médicos, pode-se utilizar a técnica de expansão para aumentar matematicamente a quantidade de registros.

## **Geração da Base de Dados de Comparação por Pares**

Considerando todas as possibilidades de registros de pacientes geradas na etapa anterior, foi possível criar um método que permitisse a compara-

ção de todas as possibilidades dentro da base de dados. Para isso, foi aplicado o método de comparação por pares utilizando técnica binária.

Esse método consiste em comparar todas as possibilidades de combinações em pares entre as variáveis dos pacientes, gerando uma base com mais de 6 bilhões de registros. Esse processo permitiu ampliar significativamente a capacidade de análise dos dados e identificar relações entre variáveis que não seriam possíveis com as limitações anteriores.

O uso dessa técnica tem se mostrado extremamente útil para a identificação de padrões em grandes conjuntos de dados. No contexto médico, pode ser utilizado para a identificação de fatores de risco e padrões de comportamento em doenças, auxiliando na elaboração de estratégias de prevenção e tratamento mais eficazes.

Em resumo, a aplicação do método de comparação por pares utilizando técnica binária permitiu a ampliação da capacidade de análise dos dados médicos, gerando uma base com 6.006.250.000 registros. Essa técnica é extremamente útil para a identificação de padrões e relações entre variáveis.

## **Geração de uma massa de dados gerenciável**

O processo de treinamento de um modelo utilizando o algoritmo *random forest* geralmente envolve a paralelização (execução de várias tarefas em paralelo) e requer alta capacidade de memória RAM envolvida para a leitura do arquivo. O processo de construção de cada árvore envolve uma busca através de todos os atributos para determinar qual é o melhor atributo para dividir o conjunto de dados em cada nó da árvore, tornando uma execução computacionalmente intensiva, especialmente em conjuntos de dados grandes.

Após uma série de experimentos realizados utilizando uma base de dados anterior contendo mais de 6 bilhões de registros, constatou-se a necessidade de um ambiente computacional mais robusto para a aplicação do algoritmo de *random forest*. Tal necessidade decorre da demanda computacional envolvida na construção de múltiplas árvores de decisão, cada uma das quais pode ser consideravelmente grande, sobretudo em casos onde existem múltiplos atributos e um grande número de observações no conjunto de dados.

Após identificar o problema, foi solucionado empregando a técnica de amostragem, a qual permitiu a criação de um arquivo com uma quantidade reduzida de pacientes selecionados de forma mais diversificada, a fim de englobar a maior parte dos dados gerados, deixando incluso os resultados da validação dos médicos voluntários.

Foram gerados dois conjuntos de dados contendo informações de pacientes. O primeiro conjunto inclui uma amostra de 5.000 pacientes, em que cada paciente é comparado com todos os outros, e os resultados com valor 0 foram removidos. Este conjunto é destinado para treinar o modelo *random forest*. O segundo conjunto também contém uma amostra de 5.000 pacientes, em que cada paciente é comparado com todos os outros, e não houve remoção de resultados com valor 0. Este conjunto será utilizado para validar o modelo treinado, por meio de validação cruzada externa aos dados de treinamento.

Com as bases de dados geradas, torna-se possível realizar novas etapas do trabalho proposto. Para tanto, é apresentado o sistema de validação dos dados, o qual permitirá avaliar a qualidade e confiabilidade das informações geradas.

### 4.3 Sistema de validação de dados

Nesta seção, é apresentada a arquitetura do *software* desenvolvido para a validação dos dados gerados na etapa anterior. Conforme proposto no trabalho, a arquitetura foi escolhida com o objetivo de otimizar o tempo, sendo construída no *framework* Django, que utiliza a arquitetura MTV (*Model, Template, View*). O uso dessa arquitetura permitiu uma melhor organização do código, separando a lógica de negócio (*Model*) da apresentação (*Template*) e da interação do usuário (*View*). Isso também facilitou a manutenção do *software*, permitindo uma melhor escalabilidade e expansibilidade para futuras implementações. O código fonte está disponível para consulta e uso [57].

A camada *Template* é a responsável pelas páginas visuais, onde o médico estará em contato direto, enviando as informações para a próxima camada. A camada *View* recebe as informações que estão sendo enviadas e as processa conforme a programação, devolvendo as informações para o *Template*. A *View* é uma camada intermediária da arquitetura, tendo contato contínuo com a camada *Model*. Por fim, a camada *Model* tem a responsabilidade de conexão entre o sistema gerenciador de banco de dados, PostgreSQL, com a camada *View*.

Na Figura 4.2, é ilustrado o fluxo de comunicação entre as camadas *Model, Template, View*. É importante destacar que é estruturado dessa maneira para separar a lógica de negócios da interface do usuário, tornando o desenvolvimento de sistemas organizado e escalável.

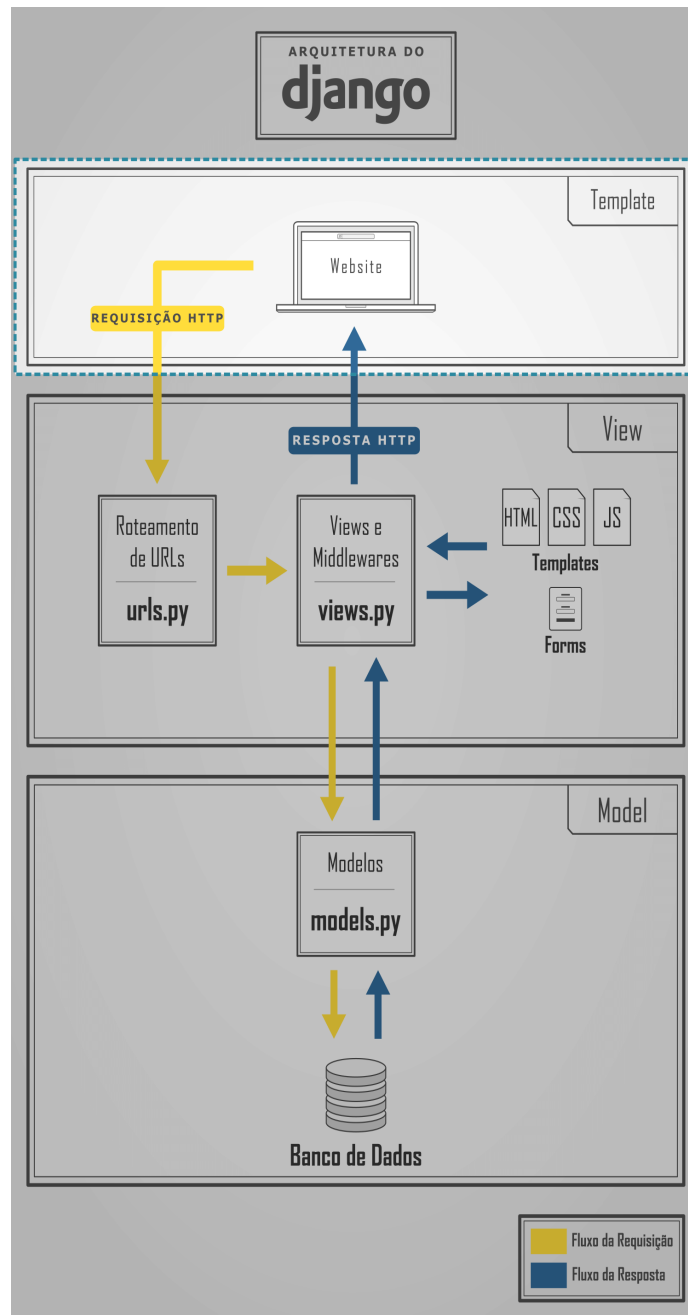


Figura 4.2: Arquitetura do Django.

Com base no *framework* Django, foi possível projetar uma estrutura para a comunicação dos dados gerados e a validação dos médicos voluntários. A Figura 4.3 apresenta de forma abrangente como as camadas da arquitetura se conectam para chegar ao resultado final. A interface, que envolve uma interação humano-computador direta com o médico, é conectada com a camada responsável pela comunicação com a base de dados, onde os dados são armazenados.

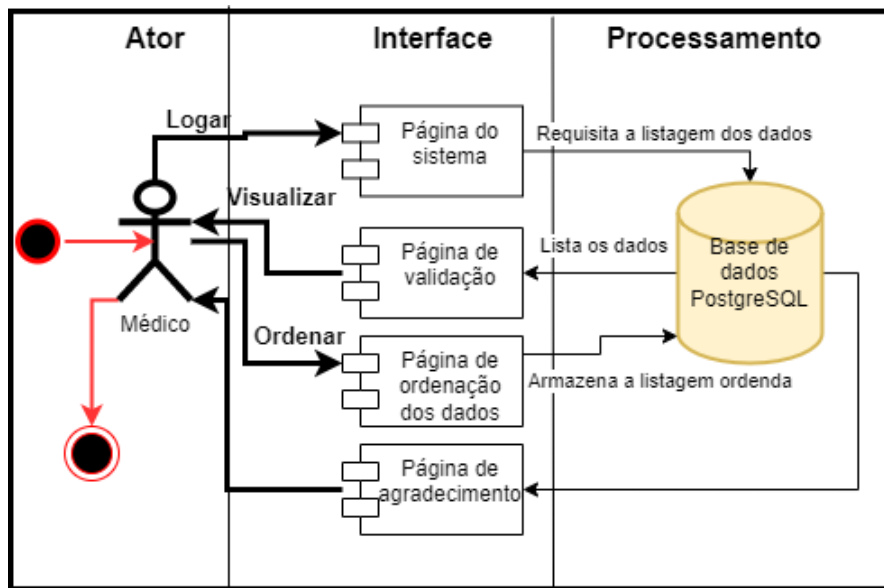


Figura 4.3: Arquitetura do sistema de validação proposto.

A base de dados é dividida em duas partes: os dados dos usuários do sistema (*login* e senha) e dados clínicos gerados. Estes últimos, por sua vez, são divididos em duas etapas: antes e depois da validação. No entanto, a arquitetura do sistema não é suficiente para concluí-lo. É necessário utilizar tecnologias para implementar as regras e modelos, bem como para desenvolver a interface, incluindo estilização e funcionalidades. Nesse sentido, sugerem-se tecnologias adequadas para essa finalidade.

## 4.4 Tecnologias utilizadas

No desenvolvimento deste trabalho, foram empregadas várias tecnologias críticas para a construção do sistema, incluindo: o *HyperText Markup Language* (HTML), que foi utilizado para estruturar e criar as páginas da Web; o

*Cascading Style Sheets* (CSS), que forneceu a estilização das páginas; e o JavaScript, uma linguagem de programação de script, que permitiu a criação de interatividade e dinamismo nas páginas da Web.

Devido à sua sintaxe fácil de ser aplicada, juntamente com uma grande comunidade que desenvolve bibliotecas e *frameworks*, o Python foi escolhido como a linguagem de programação base para esta pesquisa. O índice TIOBE confirmou o Python como a linguagem de programação mais popular do mundo nos anos de 2018, 2020 e 2021, em grande parte devido às suas várias bibliotecas de alto desempenho, que permitem a criação de modelos de inteligência artificial de forma rápida e eficiente. Python é uma linguagem multiplataforma, o que significa que os algoritmos de IA desenvolvidos em Python podem ser executados em diferentes sistemas operacionais e dispositivos, o que aumenta a portabilidade dos modelos de IA.

O sistema gerenciador de banco de dados escolhido para este estudo foi o PostgreSQL, um sistema avançado de gerenciamento de banco de dados relacional que suporta as características SQL de código aberto. O PostgreSQL foi concebido com o objetivo de fornecer uma solução segura, confiável, estável, eficiente e flexível para o armazenamento de informações. A sua flexibilidade permite que ele seja utilizado em uma ampla variedade de aplicações, incluindo aplicações de negócios, mídias sociais, inteligência artificial e análise de dados [41].

Heroku é uma plataforma de nuvem que permite aos desenvolvedores construir, implantar e gerenciar aplicativos em nuvem. Fundada em 2007, é propriedade da Salesforce usada por mais de 7 milhões de desenvolvedores em todo o mundo [40]. A plataforma foi utilizada para armazenar os dados e resultados obtidos, sendo responsável por hospedar tanto o código fonte do sistema quanto a base de dados, oferecendo uma solução de nuvem para implantação, gerenciamento e escalabilidade de aplicações web de maneira simplificada e eficiente oferecendo alta disponibilidade e segurança para os dados armazenados, o que é fundamental para aplicações médicas que lidam com informações sensíveis. Assim, utilizando o PostgreSQL como sistema gerenciador de banco de dados e a Heroku como servidor de hospedagem, garante-se a integridade e segurança dos dados, além de assegurar a disponibilidade e escalabilidade do sistema.

## 4.5 Ambiente de análise de dados

Toda a fase de geração dos dados foi realizada no computador que possui as configurações conforme a Tabela 4.1, utilizando a ferramenta Visual

Studio Code, com a instalação do Python versão 3.10.5. Já na fase de análise de dados e treinamento do algoritmo, inicialmente foi decidido utilizar a ferramenta Google Colaboratory, que tem a vantagem de ser online. Porém os arquivos começaram a ficar com tamanhos maiores (12 GB, 5 GB) e a ferramenta se limitou ao receber esses arquivos, assim como o estouro de memória RAM limitada a 12 GB. Para resolver essa questão, utilizou-se o computador descrito na Tabela 4.1 com o pacote Anaconda:

REQUISITO	DESCRIÇÃO
Sistema Operacional (SO)	Microsoft Windows 10
Processador	Intel64 Family 6 Model 158 Stepping 10 Genuine Intel 2592 Mhz
Memória RAM	8 GB
HDD	1 TB (sendo 500 GB para a geração dos dados)

**Tabela 4.1:** Configurações do computador utilizado.

Anaconda é uma distribuição de software livre e aberto voltada para ciência de dados e análise numérica. Ela inclui diversas bibliotecas, ferramentas e ambientes de desenvolvimento integrado (IDEs) pré-instalados, o que facilita o trabalho de cientistas de dados, analistas e programadores. A distribuição é desenvolvida e mantida pela empresa Anaconda Inc. e está disponível para os sistemas operacionais Windows, MacOS e Linux [39].

Algumas das principais bibliotecas que vêm pré-instaladas no Anaconda incluem NumPy, Pandas, Matplotlib, Scikit-learn, Jupyter Notebook e Spyder. Além disso, o Anaconda oferece um sistema de gerenciamento de pacotes que permite instalar facilmente outras bibliotecas e ferramentas. Bibliotecas utilizadas durante o estudo:

- **Pandas:** Fornece estruturas de dados rápidas, flexíveis e expressivas para tornar a análise de dados mais fácil e intuitiva. É amplamente utilizada na análise de dados, fornecendo funcionalidades avançadas para transformar, filtrar, agregar e visualizar dados de forma clara e eficiente, trabalha com dados em formato de tabela, seja em arquivos CSV, Excel ou bancos de dados SQL [38].
- **Seaborn:** É uma biblioteca de visualização de dados em Python que é construída sobre o Matplotlib. Ela oferece uma interface de alto nível para criar gráficos estatísticos atraentes e informativos [38].

- **Random:** Essencial para diversas aplicações científicas e tecnológicas que envolvem a modelagem de eventos aleatórios. Essa biblioteca é capaz de gerar sequências de números pseudoaleatórios com base em algoritmos determinísticos, que possuem propriedades estatísticas semelhantes às de uma sequência de números verdadeiramente aleatórios [38].
- **Collections:** Oferece uma série de estruturas de dados eficientes que são alternativas a listas, tuplas e dicionários. Essas estruturas de dados são especialmente valiosas quando se trabalha com grandes volumes de informações, pois podem melhorar significativamente o desempenho do código. A biblioteca *collections* inclui recursos adicionais afim de entregar rapidez em inserções, remoções e acesso a elementos, além de fornecer tipos de dados com estruturas específicas que podem ser mais adequadas para determinados problemas [38].
- **Sklearn.ensemble.RandomForestClassifier:** É um algoritmo de aprendizado de máquina que utiliza um conjunto de árvores de decisão aleatórias para realizar classificação. Ele é frequentemente utilizado em problemas de classificação e é conhecido por sua alta precisão e baixo *overfitting* [30].
- **Sklearn.preprocessing.LabelEncoder:** É uma classe em Scikit-learn que é usada para transformar variáveis categóricas em numéricas. Isso é útil para que algoritmos de aprendizado de máquina possam trabalhar com esses dados de forma mais eficiente [30].
- **Sklearn.model\_selection.train\_test\_split:** É uma função em *Scikit-learn* que divide um conjunto de dados em conjuntos de treinamento e teste. Isso é útil para avaliar o desempenho de um modelo de aprendizado de máquina em dados não vistos durante o treinamento [24].
- **Sklearn.metrics.confusion\_matrix:** A matriz de confusão é uma métrica usada para avaliar o desempenho de um modelo de classificação. Ela compara as previsões feitas pelo modelo com as classes reais dos dados e produz uma tabela que mostra os resultados de verdadeiros positivos, verdadeiros negativos, falsos positivos e falsos negativos. Esses resultados são úteis para avaliar a precisão e *recall* do modelo [38].
- **Csv:** A biblioteca *csv* em Python permite que um programador leia e escreva arquivos CSV (comma-separated values). É uma biblioteca padrão em Python, o que significa que é instalada com o próprio Python e não requer instalação adicional [39].

- **Itertools:** Fornece uma coleção de funções para trabalhar com iterações, que são objetos que podem ser iterados (como listas ou tuplas). Ela fornece funções úteis como combinações, permutações e produto cartesiano, que são úteis em diversas aplicações, como análise de dados e criação de jogos [39].
- **Generator:** É capaz de criar geradores, que são funções que produzem uma sequência de valores em vez de retornar uma lista de valores. Isso é útil para processar grandes conjuntos de dados, pois evita a necessidade de armazenar toda a sequência na memória [39].
- **Random:** Fornece funções para gerar números aleatórios. Ela pode ser usada para criar simulações e jogos que envolvem elementos aleatórios [39].
- **Multiprocessing:** Permite que um programador execute tarefas em paralelo, usando vários processos em vez de um único processo. Isso pode melhorar significativamente o desempenho em computadores com vários núcleos de processamento. Ela é especialmente útil em aplicações que envolvem grandes quantidades de dados ou processamento intensivo de CPU [39].

## 4.6 Resumo do capítulo

Neste capítulo, foi apresentada uma visão geral das etapas de construção, seguido pela estruturação da arquitetura computacional assim como as tecnologias envolvidas e suas aplicações. Detalhou-se como o modelo está baseado no protocolo e como se deu sua geração dos dados no sistema e suas respectivas regras.



---

# Capítulo 5

## Aplicação do Modelo

---

*Knowing is not enough; we must apply.  
Willing is not enough; we must do.*

*Johann W. von Goethe, German Novelist (1749–1832)*

**N**este capítulo são descritos a construção da aplicação do sistema de validação dos médicos voluntários, o gerenciamento dos dados aplicados ao modelo computacional, bem como a leitura e tratamento de dados, juntamente com a análise exploratória. Também foram descritos o treinamento e a validação do modelo, bem como os resultados e discussões do experimento realizado com 2 milhões de casos e dificuldades encontradas.

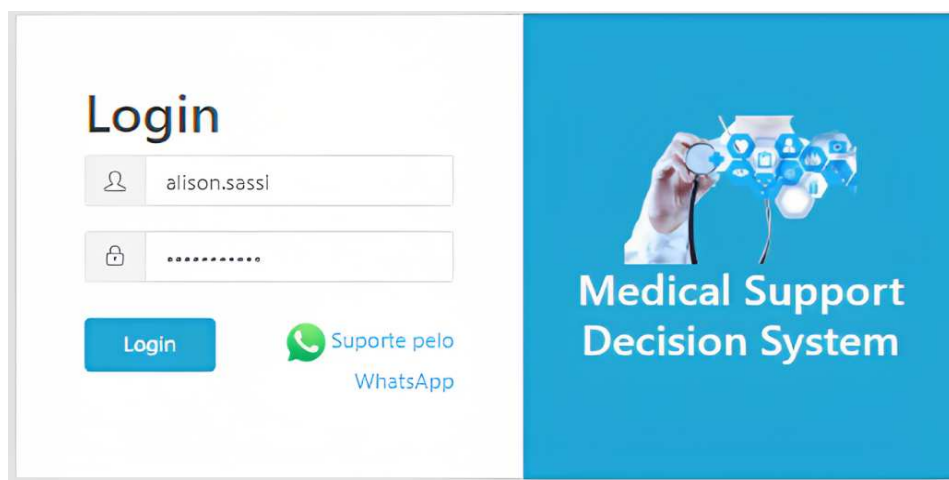
### 5.1 Descrição do experimento

Após um levantamento da necessidade dos ambientes de geração de dados, treinamento e validação do modelo, iniciou-se o processo de geração de massa de dados, na primeira análise notou-se que a partir das regras do protocolo, existe uma saída decisória para a equipe clínica quando houver casos clínicos iguais. Para realizar um experimento completo, foi desenvolvido um *software* denominado *Medical Decision Support* [57], com a intenção de extrair do conhecimento do médico voluntário, através de validação de dados que o protocolo não foi capaz de priorizar.

### 5.2 Sistema de validação

Implementou-se uma página de *login*, ilustrado na Figura 5.1, que possibilita a autenticação de usuários do sistema, os médicos, através da inserção

de nome de usuário e senha. A página foi projetada para ser intuitiva e de fácil utilização pelos usuários, e conta com medidas de segurança adicionais, tais como criptografia de senhas e verificação de tentativas de *login* suspeitas, para garantir a proteção das informações. A página foi implementada com uma funcionalidade de suporte por meio do aplicativo de mensagens *WhatsApp*, disponível para estabelecer uma comunicação entre usuários e o suporte. Os resultados de vários testes de usabilidade realizados, apontaram para de autenticação e simplicidade.

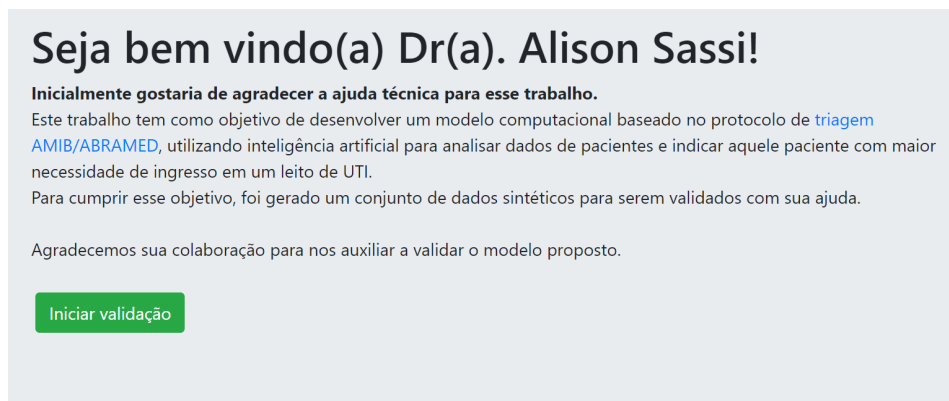


**Figura 5.1:** Tela de login do sistema de validação.

Na Figura 5.2 apresenta uma página com uma mensagem de agradecimento aos usuários pelo acesso e fornece uma explicação detalhada sobre o processo de validação que será utilizado no sistema. Esta página tem como objetivo orientar os usuários sobre os procedimentos necessários para validar suas informações e garantir a segurança e integridade do sistema.

Criou-se uma página que apresenta uma listagem de grupos aleatórios para que os usuários possam escolher um deles como ponto de partida para iniciar o processo de validação. Conforme a Figura 5.3, a listagem é gerada de forma aleatória para garantir a imparcialidade do processo e permitir que os usuários escolham o grupo que melhor se adéqua às suas necessidades e interesses.

Apresentou-se uma lista, conforme ilustrado na Figura 5.4, em uma nova página aos médicos de dez registros de pacientes hipotéticos, com os dados do protocolo expostos de forma clara. Esta página tem como objetivo fornecer aos médicos um mecanismo para avaliar e ordenar os pacientes de acordo



**Figura 5.2:** Tela de introdução ao sistema.



**Figura 5.3:** Tela da escolha dos grupos no sistema.

com seu conhecimento e experiência clínica. A página possui uma funcionalidade que permite aos médicos marcar um dos pacientes como inválido, caso considerem que os dados apresentados são incorretos ou incompletos. Após a ordenação dos pacientes, os médicos podem enviar os dados para armazenamento, assegurando assim a integridade dos registros.

Implementou-se uma funcionalidade que exige que os médicos justifiquem o motivo da marcação de um paciente como “inválido” quando utilizam esta opção, conforme Figura 5.5. Esta medida foi adotada com o objetivo de garantir a transparência e a justificabilidade das decisões dos médicos e possibilitar a discussão posterior destas decisões, caso se faça necessá-

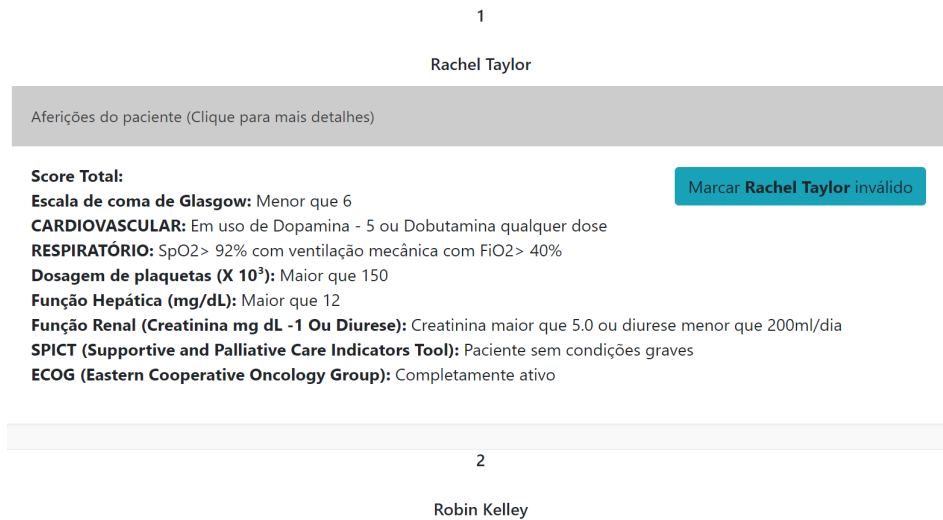


Figura 5.4: Tela de validação ao sistema.

rio. A justificativa fornecida pelos médicos é armazenada junto com os dados do paciente para fins de auditoria e avaliação da qualidade do processo.



Figura 5.5: Tela de justificativa do sistema.

Uma página de agradecimento é exibida aos médicos após o término de um processo de validação, tendo como objetivo de expressar gratidão pelo empenho e dedicação dos médicos na realização da validação e motivá-los a continuar participando destes processos de forma ativa e comprometida, ilustrado na Figura 5.6.



**Figura 5.6:** Tela de agradecimento do sistema.

Os resultados obtidos na validação do modelo com a participação de três profissionais voluntários foram limitados em sua amplitude e não alcançaram as expectativas previamente estabelecidas, foram validados vinte e oito registros de pacientes. Durante a fase de validação por médicos voluntários, houve uma dificuldade para ordenar os pacientes devido às limitações das regras e variáveis incluídas no protocolo AMIB/ABRAMEDE, que não abrangeram adequadamente o contexto e estado de saúde dos pacientes.

### **Justificativa dos médicos voluntários sobre o contexto do paciente**

Durante a fase de validação, houve um *feedback* dos médicos voluntários em relação aos dados dos pacientes e à necessidade de aprimoramento. A fim de melhorar a precisão e eficácia do modelo, esses médicos sugeriram a inclusão de variáveis adicionais, tais como idade, tempo de internação, chance de recuperação e doença de origem, a fim de contextualizar os dados dos pacientes de maneira mais adequada e aprimorar a validação do modelo.

De acordo com a opinião do entrevistado, os critérios objetivos utilizados como base para a tomada de decisão são considerados corretos, porém, insuficientes. O entrevistado argumentou que, em diversas ocasiões durante a pandemia, os critérios utilizados para a tomada de decisão são iguais e, portanto, limitados. Em sua visão, fatores como idade, condição socioeconômica e chance de recuperação (dada a gravidade do COVID-19) deveriam ser considerados para complementar e aprimorar os critérios objetivos utilizados na tomada de decisão.

Outro profissional acredita que, embora a medicina não seja uma ciência exata como a matemática, é sempre importante tentar quantificar as

informações. No entanto, acredita que a intuição profissional e as considerações éticas fazem parte da humanidade e não devem ser ignorados. Por isso, a pessoa acredita que questões mais “humanas” e filosóficas podem ser aplicadas para lidar com nuances e diferenciações na medicina.

Todos os médicos informaram que a variável idade é primordial para a decisão médica. A pessoa entrevistada destacou a importância da idade como fator indivisível em relação ao paciente. Apesar de não ter sido incluída no protocolo utilizado, a idade pode influenciar significativamente a saúde, tanto no diagnóstico quanto no prognóstico, na recuperação e na chance de sobrevivência. Em um cenário de calamidade pública e recursos limitados, a idade pode ser um fator importante a ser considerado. Em situações em que dois pacientes com alto risco de vida disputam um único leito, o mais jovem pode ser preferido, pois tem maior chance de sobreviver se receber o leito. No entanto, se os dois pacientes têm alto risco de vida, mas sem sinais de deterioração, o mais idoso pode ser preferido, pois tem maior necessidade do leito e menos chance de sobreviver sem ele, enquanto o jovem tem chance de sobreviver sem o leito. A idade é, portanto, um fator importante a ser considerado ao avaliar o risco e a sobrevivência dos pacientes em cenários de escassez de recursos.

Segundo o depoimento de um profissional com 10 anos de experiência, a inclusão do tempo de espera na fila é um dado que poderia ser considerado, embora não seja o único fator determinante. À medida que o tempo passa, as chances de sobrevivência diminuem e a necessidade aumenta. Quando um profissional de saúde se depara com um paciente que já apresenta sinais de deterioração, o paciente que aguardou mais tempo na fila necessita de cuidados adicionais e deve ser considerado prioritário.

O profissional relembrou do cenário que o trabalho se aplica, e na prática a distância é um fator relevante a ser considerado quando se trata da distribuição de leitos em um país tão extenso quanto o Brasil. De acordo com o médico, o tempo de deslocamento está relacionado à distância, e a sobrevivência do paciente está relacionada ao tempo. O médico sugere que, em caso de dois pacientes com alto risco, o paciente mais próximo ao leito deve ser priorizado, pois pode não ter condições de ser deslocado por tanto tempo. Já em relação a dois pacientes com risco intermediário, o leito poderia ser alocado ao paciente que estivesse mais tempo na fila, já que ambos têm condições de transporte.

Apesar das justificativas apresentadas pelos médicos para contextualizar os dados dos pacientes, considerando variáveis como idade, tempo de internação e doença de origem, a fim de aprimorar a validação do modelo, o

objetivo original deste trabalho é aplicar as recomendações da AMIB/ABRAMEDE em tempos de escassez. Essas associações propuseram um protocolo validado que inclui um modelo de triagem baseado em aceitabilidade social e ajuste às leis do país, abrangendo as áreas éticas, legais e de aceitação social. Esse modelo protege contra potenciais questionamentos jurídicos, é complexo o suficiente para oferecer segurança preditiva e ser aplicável na prática, e ajuda o profissional de saúde a tomar decisões complexas sobre a alocação de leitos de UTI e ventiladores, tornando o processo de decisão mais transparente.

## 5.3 Gerenciamento dos dados

Nesta seção, aborda-se em detalhes como os dados foram aplicados no modelo computacional. É importante entender que a qualidade dos resultados obtidos pelo modelo depende diretamente da qualidade dos dados utilizados. Portanto, aborda-se os processos de limpeza, pré-processamento e transformação dos dados para torná-los adequados para o modelo. São discutido os métodos utilizados para dividir os dados em conjuntos de treinamento e teste, bem como a estratégia de validação utilizada para avaliar o desempenho do modelo.

### Preparação de dados gerados

A etapa de preparação de dados é um processo importante para garantir que os dados utilizados em análises e modelagem estejam em um formato adequado e pronto para serem processados de forma eficiente. Essa fase engloba diversas etapas, desde a identificação e tratamento de dados ausentes até a análise da quantidade e distribuição dos dados, bem como a seleção e transformação de variáveis relevantes para a análise em questão. Uma preparação inadequada pode resultar em resultados imprecisos e comprometer a qualidade da análise realizada. Por isso, é fundamental que as etapas de preparação de dados sejam conduzidas com rigor e atenção aos detalhes.

Nas etapas de geração de dados explicados anteriormente, houve-se um tratamento e validação prévio, houveram vários experimentos durante este processo, alterando a massa de dados como também a preparação. Agora, são descritas as etapas de importação e preparação dos dados, fundamentais para a execução do treinamento do modelo, será inicialmente explicado o processo aplicado para os experimentos com dados provenientes de um experimento específico.

## Leitura dos dados

O arquivo final com os dados possui 22 colunas com mais de 6 bilhões de linhas. O arquivo está no formato CSV (*Comma-separated values*) registrando o tamanho de 291 GB, na Figura 5.7 tem uma amostra da visualização dos dados.

0_vs_0,2,4,0,1,4,3,0,0,11,2,2,4,0,1,4,3,0,0,11,2,0
0_vs_1,2,4,0,1,4,3,0,0,11,2,2,3,2,2,2,0,0,0,11,2,0
0_vs_2,2,4,0,1,4,3,0,0,11,2,4,0,4,0,0,4,3,0,8,4,1
0_vs_3,2,4,0,1,4,3,0,0,11,2,1,3,1,0,4,1,3,0,9,5,1
0_vs_4,2,4,0,1,4,3,0,0,11,2,0,4,0,0,1,0,0,0,5,1,2
0_vs_5,2,4,0,1,4,3,0,0,11,2,0,1,1,4,2,1,0,0,8,1,2
0_vs_6,2,4,0,1,4,3,0,0,11,2,4,0,3,2,3,1,0,0,12,3,1
0_vs_7,2,4,0,1,4,3,0,0,11,2,0,1,1,0,3,2,3,0,5,4,1
0_vs_8,2,4,0,1,4,3,0,0,11,2,0,4,2,0,1,3,0,0,7,1,2
0_vs_9,2,4,0,1,4,3,0,0,11,2,1,1,2,4,4,1,0,0,12,3,1
1_vs_0,2,3,2,2,2,0,0,0,11,2,2,4,0,1,4,3,0,0,11,2,0
1_vs_1,2,3,2,2,2,0,0,0,11,2,2,3,2,2,2,0,0,0,11,2,0
1_vs_2,2,3,2,2,2,0,0,0,11,2,4,0,4,0,0,4,3,0,8,4,1

Figura 5.7: Amostra dos dados no arquivo CSV.

Ao importar um arquivo CSV em Python, é crucial especificar a codificação de caracteres correta para garantir a leitura e processamento adequados dos dados. No entanto, durante a importação do arquivo, as restrições técnicas do ambiente computacional impediram sua importação. Para superar essa limitação, foi necessário criar um novo arquivo CSV com tamanho reduzido de 1,35 GB, o que tornou possível sua importação. No entanto, para lidar com a quantidade de dados, é necessário utilizar técnicas de importação de dados seletivos e exigir uma menor alocação de memória, como demonstrado no Código 5.1.

Listing 5.1: Leitura e codificação dos dados

```
1 data_origin = pd.read_csv(r'\Data_Generate.csv',
    encoding='latin-1', skiprows=lambda i: i>0 and
    random.random() > 0.80)
```

Utiliza-se o parâmetro `encoding='latin-1'` na importação do arquivo, pois ele especifica a codificação de caracteres do arquivo, o valor `latin-1` é uma codificação de caracteres que é amplamente utilizada em países de línguas ocidentais, incluindo os países de língua portuguesa. Ela é

uma codificação de 8 bits que pode representar até 256 caracteres diferentes e é compatível com a maioria dos sistemas de computadores [47].

Utiliza-se também, o argumento `skiprows` para tratar a quantidade de registros no arquivo, usando uma função *lambda*. O parâmetro `i` o que representa o número da linha do arquivo que está sendo lida. A biblioteca *Random* do Python é utilizada para decidir aleatoriamente se uma linha deve ser pulada ou não. A condição `i > 0` garante que a primeira linha, que geralmente contém os cabeçalhos das colunas, não será pulada, e como o CSV gerado não possui cabeçalho, ela deve ser lida do arquivo [64].

A expressão `random.random() > 0.05` gera um número aleatório entre 0 e 1 e verifica se é maior que 0,05 (ou seja, se tem 95% de chance de ser verdadeira). Se essa condição for verdadeira, a linha atual será pulada e não será incluída na leitura do arquivo. Caso contrário, a linha será incluída [64].

Portanto esse parâmetro com a função *lambda* é uma maneira encontrada de ler um arquivo CSV com uma taxa de amostragem aleatória de 80%. Isso pode ser útil para reduzir o tamanho do arquivo e melhorar o desempenho da análise de dados, especialmente quando o arquivo contém uma grande quantidade de dados.

## Tratamento dos dados

Dando seguimento à transformação dos dados, utiliza-se os Código 5.2:

Listing 5.2: Remoção da coluna "patients"

```
1 data = data_origin.drop('patients', axis=1)
```

É comum que exista colunas nos dados que não contribuam para o treinamento, na linha 1 do Código 5.2 utiliza-se a função `drop()`, fornecida pela biblioteca *Pandas*, que permite remover colunas, no caso a coluna `patients`, do *DataFrame*.

A coluna `patients` trata dos dados identificadores de cada comparação de paciente, ela é removida do *DataFrame* para que não seja treinada como variável, mas sim como *label* do treinamento. Usa-se o segundo argumento `axis` que diz a respeito, do eixo ao longo das colunas do qual a operação de remoção deve ser realizada.

Listing 5.3: Conversão do tipo de dados

```
1 data = data.astype('int16')
```

O tipo de dados em cada coluna dos dados lidos, são do tipo `int64` que representa números inteiros de 64 bits e ocupa 8 bytes na memória. Isso significa que ele pode representar um total de  $2^{64}$  valores diferentes, que variam de  $-9223 \times 10^{15}$  a  $9223 \times 10^{15}$ . Portanto decidiu-se converter o tipo de dados de todas as colunas com a função `astype()` para o tipo `int16` que são números inteiros de 16 bits e ocupa 2 bytes na memória. Isso significa que ele pode representar um total de  $2^{16}$  valores diferentes, que variam de -32.768 a 32.767.

Não ocorreram muitas modificações nos dados, devido à validação prévia na fase da geração da base de dados seguindo conforme o protocolo AMIB/ABRAMEDE. No próximo ponto deste estudo, realiza-se uma explicação da análise detalhada dos dados de cada experimento registrado considerado relevante. Essa análise envolverá o processamento das informações dos dados selecionados e a aplicação de técnicas de análise exploratória para identificar padrões, tendências e relações entre as variáveis do experimento.

## Análise exploratória de dados

Na etapa de análise exploratória de dados, serão utilizadas diversas técnicas estatísticas, como a análise de medidas de centralidade, dispersão e correlação, além da visualização dos dados por meio de gráficos e outras ferramentas. Essas técnicas permitirão a identificação de padrões ocultos e relações entre as variáveis do experimento, o que pode levar à descoberta de novas hipóteses e *insights* [38].

Para iniciar a análise, é necessário compreender a quantidade e distribuição dos registros na base de dados tratada. Para isso, o Código 5.4 foi criado:

Listing 5.4: Mapeamento de registros

```
1 print(len(data.index))
2 print(data['decisao'].value_counts())
3 data['decisao'].value_counts().plot(kind='pie',
    autopct='%0.2f%%', label = 'Decisao')
```

Conhecer o tamanho do conjunto de dados é importante para garantir que a análise seja realizada de forma adequada e precisa, além de permitir uma avaliação do desempenho do modelo em relação ao tamanho do conjunto de dados utilizado.

Uma operação comum na análise de dados após importar um *Dataset* é executar o código da linha 1, que permite obter o número de linhas ou

observações de um conjunto de dados. Outra forma de visualizar os dados é verificar a contagem de ocorrências de cada valor da coluna `decisao` (0, 1 e 2), utilizando método `value_counts()`, permite um retorno em que cada índice representa um valor único presente na coluna `decisao` e o valor associado a cada índice corresponde ao número de ocorrências desse valor na coluna.

Na linha 3 do Código 5.4 cria um gráfico de pizza, que é utilizado para representar a distribuição de frequência da coluna `decisao`. A função `value_counts()` é utilizada para contar o número de ocorrências de cada valor único na coluna `decisao`. O método `plot()` é utilizado para gerar a imagem do gráfico de pizza, onde `kind='pie'` especifica o tipo de gráfico a ser criado. O parâmetro `autopct` é utilizado para formatar o valor mostrado em cada fatia da pizza, e por fim, o parâmetro `label` é utilizado para especificar o rótulo do gráfico [38].

O gráfico de dispersão é uma técnica estatística que permite analisar a relação entre duas variáveis contínuas [38]. No Código 5.5, é gerado um gráfico de dispersão para cada par de variáveis no conjunto de dados, o que permite visualizar a distribuição conjunta de cada par de variáveis e sua relação.

Listing 5.5: Gráfico de dispersão

```
1 plt.figure(dpi=300)
2 sns.pairplot(data.sample(n=100), corner=True)
```

Este código utiliza a biblioteca *Seaborn*, que é uma biblioteca de visualização de dados para Python, e a função `pairplot()` desta biblioteca serve para criar um gráfico de dispersão das variáveis, gera também um histograma das variáveis na diagonal do gráfico, o que ajuda a visualizar a distribuição de cada variável individualmente [37].

O argumento `data.sample(n=100)` é uma maneira de selecionar aleatoriamente cem observações do conjunto de dados e criar o gráfico apenas com essas observações. Isso pode ser útil quando o conjunto de dados é muito grande e a visualização de todas as observações pode ser muito complexa. Utiliza-se o argumento `corner=True`, que especifica que apenas os gráficos da diagonal superior direita do gráfico serão exibidos. Isso economiza espaço e evita redundância, uma vez que as outras informações podem ser obtidas no reflexo dos gráficos de dispersão.

O Código 5.6 descrito cria um mapa de calor que representa a matriz de correlação das variáveis contidas no conjunto de dados. Cada célula

do mapa de calor é preenchida com uma cor que varia em intensidade conforme o valor da correlação entre as variáveis correspondentes, sendo que tons mais claros indicam uma correlação positiva mais forte e tons mais escuros indicam uma correlação negativa mais forte.

Listing 5.6: Métrica Matrix de Correlação

```
1 correlation_matrix = data.corr()
2 sns.heatmap(correlation_matrix, annot=True,
              cmap='coolwarm')
```

A função `dataset.corr()` é utilizada para calcular a matriz de correlação das variáveis do conjunto de dados. A correlação é medida através do coeficiente de correlação de Pearson, que é um número entre -1 e 1 que indica a força e a direção da relação linear entre duas variáveis. Se duas variáveis têm uma correlação positiva, elas se movem na mesma direção, enquanto que uma correlação negativa indica que elas se movem em direções opostas [38]. A função `sns.heatmap()` da biblioteca Seaborn é utilizada para criar o mapa de calor a partir da matriz de correlação. O argumento `cmap='coolwarm'` define uma paleta de cores que varia de azul escuro para vermelho escuro. A diagonal principal do mapa de calor apresentará valores iguais a 1, pois representa a correlação de uma variável com ela mesma, que é sempre perfeita.

A visualização por meio do mapa de calor é uma ferramenta útil para explorar a relação entre as variáveis do conjunto de dados, permitindo identificar padrões e *insights* relevantes que possam ser úteis para a análise, porém por se tratar de 20 variáveis neste estudo, não é legível quando a figura está de forma estática, por esse motivo não coloca-se nos resultados.

## Treinamento do modelo

A etapa de separação dos dados de treino e teste e treinamento do algoritmo é uma etapa fundamental na análise de dados e no desenvolvimento de modelos de aprendizado de máquina. Essa etapa envolve a divisão do conjunto de dados disponível em duas partes: um conjunto de dados para treinamento e outro conjunto de dados para teste de validação cruzada do modelo. Uma vez que o modelo é treinado, ele pode ser utilizado para fazer previsões sobre novos dados ou para entender melhor as relações entre as variáveis. Para a etapa de separação de dados criou-se o código 5.7:

Listing 5.7: Atribuição de dados a cada variável

```
1 X = data.drop('decisao', axis=1)
2 Y = data['decisao'].values.tolist()
```

Na linha 1 do Código 5.7, a variável `X` é criada a partir da exclusão da variável `decisao` do conjunto de dados original. A separação da variável `X` envolve a criação de uma matriz de recursos que serão utilizadas para treinar o algoritmo de *random forest*.

A variável `Y` deve conter os rótulos das classes que serão previstas pelo modelo, na linha 2 do código, é criado a partir da seleção da variável `decisao` do conjunto de dados original, utilizando a função `values.tolist()` da biblioteca `Pandas`. A separação das variáveis é fundamental para garantir que o modelo seja treinado apenas com as variáveis relevantes para a tarefa de predição.

Listing 5.8: Separação de dados para treinamento e teste

```
1 X_train, X_test, Y_train, Y_test = train_test_split(X,
    Y, test_size=0.3)
```

O Código 5.8 apresentado é responsável pela separação dos dados em conjuntos de treinamento e teste utilizando a função `train_test_split` da biblioteca `Sklearn.model_selection`. Essa função recebe como entrada as variáveis `X` e `Y` que correspondem, respectivamente, às variáveis independentes (ou explicativas) e à variável dependente (ou resposta) do conjunto de dados. A função também recebe um parâmetro `test_size` que define a proporção dos dados que serão utilizados para teste (no caso, 30%), a escolha da proporção de dados para treinamento e teste é importante e pode afetar significativamente a qualidade do modelo.

Quatro conjuntos de dados são retornados: `X_train` e `Y_train`, que correspondem aos dados de treinamento, e `X_test` e `Y_test`, que correspondem aos dados de teste. A variável `X_train` contém uma parte dos dados da variável `X` utilizada para treinar o modelo, enquanto a variável `X_test` contém uma parte dos dados da variável `X` que serão usados para testar o modelo. De maneira análoga, as variáveis `Y_train` e `Y_test` contêm os valores correspondentes à variável resposta (ou variável dependente) que serão usados para treinar e testar o modelo.

Listing 5.9: Treinamento do modelo com parâmetros

```
1 clf_RF = RandomForestClassifier(n_estimators=100,
    random_state=42)
2 clf_RF = clf_RF.fit(X_train, Y_train)
```

Usando a classe `RandomForestClassifier` é criado um objeto de classificação de *random forest* usando a biblioteca *Scikit-Learn* do Python. A classe é inicializada com dois parâmetros:

- `n_estimators` é um hiperparâmetro do algoritmo de *random forest* que indica o número de árvores de decisão que serão construídas para compor o modelo. Foram especificadas 100 árvores o que geralmente é suficiente para obter um modelo com bom desempenho em muitos casos.
- `random_state` é um hiperparâmetro do algoritmo de *random forest* (e de outros algoritmos de aprendizado de máquina) que controla a aleatoriedade durante o processo de treinamento do modelo. O valor 42 foi utilizado como valor de semente aleatória (`random_state`), que é um valor comumente usado na comunidade de ciência de dados.

O Código 5.9 da linha 2, é usado para treinar o modelo, utilizando o método `fit()`, que é chamado para fazer o ajuste do modelo nos dados de treinamento. Aprender a relação entre as variáveis independentes e dependentes do conjunto de dados de treinamento. A saída do método `fit()` é um modelo treinado que pode ser usado para fazer previsões sobre novos dados. A variável `X_train` representa as variáveis independentes do conjunto de dados de treinamento e a variável `Y_train` representa a variável dependente que estamos tentando prever. A variável `clf_RF` armazena o modelo treinado, permitindo que seja usado posteriormente para fazer previsões sobre dados de teste ou novos dados.

Listing 5.10: Importância das variáveis

```
1 features = data
2 features_importance = zip(clf_RF.feature_importances_,
3                           data)
3 for importance, feature in sorted(features_importance,
4                                   reverse=True):
4     print("%s: %f%" % (feature, importance*100))
```

A análise da importância das *features* é uma etapa importante na avaliação da qualidade do modelo, pois permite identificar as variáveis mais relevantes para a classificação e descartar variáveis que não contribuem significativamente para a precisão do modelo.

Com o Código 5.10 apresentado, é realizada uma análise da importância das *features* na classificação realizada pelo algoritmo de *random forest*. É

gerada uma lista contendo os valores de importância e nome de cada *features*, ordenada em ordem decrescente de importância e é exibida em tela com a porcentagem correspondente a cada *features*. A importância de cada *features* é calculada como uma medida da relevância da *features* na classificação. Essa medida é baseada no ganho de informação que a *features* fornece em cada nó da árvore de decisão do *random forest* [38].

## Validação do modelo

A fase de validação do modelo é uma etapa crucial do processo de aprendizado de máquina, que tem como objetivo avaliar o desempenho do modelo em relação a dados não vistos durante o treinamento. Essa etapa é responsável por verificar se o modelo é capaz de generalizar para novos dados e, assim, ser útil na resolução de problemas do mundo real.

Uma das principais métricas utilizadas para avaliar o desempenho do modelo é a acurácia, que representa a proporção de previsões corretas em relação ao total de previsões realizadas pelo modelo. Além da acurácia, existem outras métricas que podem ser utilizadas para avaliar o desempenho do modelo, como a precisão, o *recall* e a F1-score, que levam em conta diferentes aspectos das previsões do modelo, como falsos positivos, falsos negativos e equilíbrio entre precisão e *recall* [38].

Existem vários métodos para validar o desempenho do modelo, incluindo a validação cruzada, que consiste em dividir os dados em vários conjuntos de treinamento e teste, e a curva de aprendizado, que permite avaliar como o desempenho do modelo varia com o tamanho do conjunto de treinamento. Além disso, é importante ter cuidado ao escolher a métrica de avaliação do modelo, já que diferentes métricas podem levar a diferentes considerações sobre o desempenho do modelo. Por isso, é importante entender as limitações e as implicações de cada métrica e escolher aquela que melhor se adequar ao problema em questão [38].

Existem vários métodos cientificamente válidos para avaliar o desempenho de modelos de aprendizado de máquina, incluindo a validação cruzada, que é uma das técnicas mais utilizadas, que consiste em dividir o conjunto de dados em  $k$  subconjuntos de tamanho igual. Em seguida, é feita a avaliação do modelo  $k$  vezes, sendo que em cada iteração um dos subconjuntos é utilizado para validação e os demais para treinamento. Ao final, é calculada a média das métricas de desempenho obtidas em cada iteração, o que permite uma avaliação mais precisa e robusta do modelo [38].

Calcular a acurácia do modelo de *random forest* em relação aos dados de teste da mesma base de dados é comum na área de ciência dos dados. A acurácia é uma medida comumente utilizada para avaliar o desempenho de modelos de aprendizado de máquina, e representa a proporção de previsões corretas em relação ao total de previsões realizadas pelo modelo. Neste estudo utilizou-se o seguinte código:

Listing 5.11: Métrica de Acurácia

```
1 accuracy = clf_RF.score(X_test, Y_test)
2 print('Accuracy: ' + str(accuracy))
```

Para calcular a acurácia, o Código 5.11 utiliza a função `score()`, que recebe como entrada os dados de teste (`X_test`) e as respectivas classes verdadeiras (`Y_test`), e retorna a acurácia do modelo em relação a esses dados.

Se o modelo de *random forest* alcançou 100% de acurácia em um conjunto de dados de teste, significa que o modelo foi capaz de prever corretamente todas as amostras do conjunto de dados. Em outras palavras, o modelo não cometeu nenhum erro na previsão das classes para as amostras do conjunto de dados de teste.

O modelo atingir 100% de acurácia nem sempre significa que o modelo é perfeito e pode ser generalizado para outras amostras fora do conjunto de dados de teste. É possível que o modelo tenha sido superajustado (*overfitting*) ao conjunto de dados de treinamento, memorizando as amostras e os rótulos em vez de aprender a relação subjacente entre as variáveis e a classe.

Adicionalmente à acurácia, elaborou-se um Código 5.12 para obter uma compreensão mais abrangente do desempenho do modelo aplicado nos dados separados inicialmente:

Listing 5.12: Métricas adicionais

```
1
2 from sklearn.metrics import accuracy_score,
  precision_score, recall_score, f1_score, roc_auc_score,
  confusion_matrix
3
4 y_pred = clf_RF.predict(X_test)
5
6 precision = precision_score(Y_test, y_pred,
  average='macro')
7 recall = recall_score(Y_test, y_pred, average='macro')
```

```
8 f1 = f1_score(Y_test, y_pred, average='macro')
9 auc_roc = roc_auc_score(Y_test,
    clf_RF.predict_proba(X_test), multi_class='ovr')
10 cm = confusion_matrix(Y_test, y_pred)
```

Este código calcula várias métricas de avaliação de desempenho para um modelo de classificação de *random forest*. A precisão, *recall* e *F1-score* são medidas que avaliam a performance do modelo em prever cada classe. O *roc\_auc\_score()* é uma medida da capacidade do modelo de distinguir entre as classes e é útil quando as classes não estão balanceadas. A *confusion\_matrix()* é utilizada para avaliar o desempenho do modelo, mostrando o número de previsões corretas e incorretas para cada classe em uma matriz [38].

Cada métrica fornece uma perspectiva diferente sobre como o modelo está realizando suas previsões. Aqui estão as explicações detalhadas de cada métrica usada no código:

- **Precisão (*Precision*):** É a proporção de previsões corretas positivas em relação ao total de previsões positivas. É calculada dividindo o número de verdadeiros positivos (TP) pelo número total de previsões positivas (TP + Falsos Positivos (FP)). É uma medida importante em casos onde falsos positivos são indesejáveis, como em diagnósticos médicos.
- **Recall (Sensibilidade ou Revocação):** É a proporção de previsões corretas positivas em relação ao total de casos positivos. É calculada dividindo o número de TP pelo número total de casos positivos (TP + Falsos Negativos (FN)). É uma medida importante em casos onde falsos negativos são indesejáveis, como em diagnósticos médicos [7].
- **F1-Score:** É a média harmônica entre precisão e *recall*. É uma medida que leva em consideração ambas as métricas. É uma medida útil quando há um compromisso entre precisão e *recall* [7].
- **AUC-ROC (Area Under the Receiver Operating Characteristic Curve):** É uma medida da capacidade do modelo de distinguir entre classes positivas e negativas. É calculada plotando a curva ROC e medindo a área sob a curva. Uma pontuação de AUC-ROC de 1,0 indica um modelo perfeito, enquanto que um valor de 0,5 indica que o modelo é tão bom quanto um modelo aleatório.

- **Matriz de Confusão:** É uma tabela que mostra o número de previsões corretas e incorretas para cada classe. É uma ferramenta útil para visualizar o desempenho do modelo em cada classe e identificar onde o modelo está tendo mais dificuldade [7]. A matriz de confusão é composta pelos seguintes elementos:
  - ◊ **Verdadeiros Positivos (TP):** Observações positivas que foram corretamente classificadas como positivas pelo modelo.
  - ◊ **Falsos Positivos (FP):** Observações negativas que foram incorretamente classificadas como positivas pelo modelo.
  - ◊ **Verdadeiros Negativos (TN):** Observações negativas que foram corretamente classificadas como negativas pelo modelo.
  - ◊ **Falsos Negativos (FN):** Observações positivas que foram incorretamente classificadas como negativas pelo modelo.

Com base nos dados coletados durante o experimento, a tabela 5.1 apresenta as métricas calculadas para avaliar o desempenho do modelo testado. Essas métricas foram selecionadas para fornecer uma análise melhor e confiável sobre o desempenho do modelo em questão:

MÉTRICA	RESULTADO		
Acurácia	99.98%		
Precisão	99.99%		
Recall	99.99%		
F1-Score	1.00		
AUC-ROC	1.00		
Matriz de Confusão	26	0	0
	0	51820	7
	0	0	52279

**Tabela 5.1:** Visualização dos resultados das métricas.

## Validação do modelo

Com o algoritmo de *random forest* e os dados gerados de forma clara para que o modelo aprenda magnificamente bem, tem uma tendência a atingir 100% das métricas explicadas anteriormente. Com os resultados pode-se

perceber que o modelo atingiu um valor próximo do ideal, sendo possível que o modelo tenha sido superajustado (*overfitting*) ao conjunto de dados de treinamento, memorizando as amostras e os rótulos em vez de aprender a relação subjacente entre as variáveis e a classe. Diante disso, pensou-se em avaliar o desempenho do modelo com um conjunto de dados diferente e utilizar as mesmas métricas de avaliação.

A validação do desempenho de um modelo de *machine learning* pode ser além do que as métricas já citadas, foi realizado neste estudo a técnica validação fora da amostra, utilizada em *Data Science* para avaliar a capacidade de generalização de um modelo. A ideia é testar o modelo em um conjunto de dados que não foi utilizado para treiná-lo, simulando assim uma situação em que o modelo é utilizado em dados novos, não vistos anteriormente.

A validação externa é importante para ter uma confirmação do desempenho do treinamento da inteligência artificial e validar os dados externos importados.

Listing 5.13: Leitura dos dados e preparação

```
1 Data_validation =
  pd.read_csv(r'\training_data_reduce_7_01.csv',
             encoding='latin-1')
2 Data_validation.columns=['patients', 'neuro_1',
  'cardi_1', 'respi_1', 'coagu_1', 'hepat_1',
  'renal_1', 'icc_1', 'ecog_1', 'SOFA_total_1',
  'amib_total_1', 'neuro_2', 'cardi_2', 'respi_2',
  'coagu_2', 'hepat_2', 'renal_2', 'icc_2', 'ecog_2',
  'SOFA_total_2', 'amib_total_2', 'decisao']
3
4 Y_validation =
  Data_validation['decisao'].values.tolist()
5 Data_prepared =
  Data_validation.drop(['patients', 'decisao'], axis=1)
```

O Código 5.13 desenvolvido tem o objetivo de ler os dados um arquivo CSV e preparar para a validação do modelo de aprendizado de máquina. Cria-se um objeto chamado `Data_validation` que contém os dados do arquivo CSV, é renomeado as colunas para melhor manipulação dos dados e, por fim, removendo as colunas `patients` e `decisao` utilizando o método `drop` da biblioteca `Pandas`. O novo objeto, `Data_prepared`, será utilizado posteriormente para treinar e avaliar o modelo de aprendizado de máquina.

Listing 5.14: Avaliação dos dados

```

1  New_Predictions = clf_RF.predict(Data_prepared)
2  count = 0
3  diff = []
4  for i in range(len(Y_validation)):
5      if Y_validation[i] != New_Predictions[i]:
6          count += 1
7          diff.append((i, Y_validation[i],
8                      New_Predictions[i]))
9
10     print("Numero de elementos diferentes: ", count)
11     print("Porcentagem de diferenca: ",
12           (count/len(Y_validation))*100, "%")
13
14     c = collections.Counter((d[1], d[2]) for d in diff)
15     print('porcentagem erro grave', sum((c[(2,1)],
16     c[(1,2)]))/len(Y_validation)*100, '%')
17     print('porcentagem decisao', sum((c[0, 1],
18     c[0,2]))/len(Y_validation)*100, '%')
19     print(c)

```

Afim de fornecer uma avaliação quantitativa da performance do modelo de *random forest* em relação aos dados de validação, permitindo identificar possíveis pontos fracos do modelo e fornecendo informações úteis para sua otimização. O modelo é utilizado para gerar previsões para os dados de validação e em seguida as previsões são comparadas com os valores verdadeiros dos dados de validação para avaliar a acurácia do modelo.

Após a geração das novas previsões para o conjunto de dados de validação, um contador é inicializado e uma lista vazia `diff` é criada para armazenar as diferenças entre as previsões geradas pelo modelo e os valores verdadeiros dos dados de validação. O código, então, itera por cada elemento dos dados de validação e verifica se a previsão gerada pelo modelo é diferente do valor verdadeiro correspondente. Caso haja diferença, o contador `count` é incrementado e um registro é adicionado à lista `diff` contendo o índice do elemento, o valor verdadeiro e a previsão gerada pelo modelo.

Por fim, o código imprime o número de elementos diferentes, a porcentagem de diferença em relação ao número total de elementos e as estatísticas das diferenças encontradas, incluindo a porcentagem de erros graves e de decisão. Essas estatísticas são calculadas utilizando a biblioteca `collections.Counter` para contar o número de ocorrências de cada par de valores verdadeiros e previsões nas diferenças registradas na lista `diff`.

Realiza-se novas análises para compreender as distribuições dos dados, como já mencionado anteriormente. Após concluir a análise dos dados, constatou-se que o modelo pode ser exportado. O Código 5.15 pode ser utilizado:

Listing 5.15: Exportação do modelo criado em arquivo

```
1 pd.to_pickle(clf_RF, r'D:\Model_MDS.pickle')
```

O `pd.to_pickle()` é uma função da biblioteca Pandas que salva um objeto em um formato de arquivo de *pickle* do Python que permite salvar objetos em um arquivo binário. O armazenamento do treinamento da inteligência artificial, deve ser realizado serializando os objetos complexos no formato *pickle*, os objetos são salvos a partir do modelo de aprendizado de máquina que foi treinado previamente e está armazenado na memória do computador, incluem listas, dicionários, matrizes ou, no caso desse estudo, modelos de aprendizado de máquina. Isso possibilita a carga destes objetos em outro algoritmo ou sistema.

## Aplicação do modelo

Obtendo o arquivo no formato *pickle*, é possível carregá-lo em memória e executar um processamento em um conjunto de dados e classificá-los em ordem de prioridade. Para isso desenvolveu-se o algoritmo 5.16:

Listing 5.16: Leitura do modelo criado e o ranking

```
1 import pandas as pd
2 import sklearn
3 Modelo = pd.read_pickle('/Model_MDS_.pickle')
4 def processing_patients(*patients):
5     if len(patients) < 2:
6         print(patients)
7         return tuple([patients[i] for i in range(0,
8             len(patients))
9
10
11     pairs = [(patients[i], patients[j]) for i in
12         range(len(patients)) for j in range(i+1,
13             len(patients))]
14
15     results = []
16     for patient1, patient2 in pairs:
17         data_prepared = [y for x in [patient1[1:],
18             patient2[1:]] for y in x]
```

```

14         comparation = Modelo.predict([data_prepared])
15         if comparation == 1:
16             results.append((patient1, 1))
17         else:
18             results.append((patient2, 1))
19     unique_results = []
20     for r in results:
21         if r[0] not in unique_results:
22             unique_results.append(r[0])
23     ranked_results = []
24     for p in patients:
25         total_points = 0
26         for r in results:
27             if r[0] == p:
28                 total_points += r[1]
29         ranked_results.append((p, total_points))
30     ranked_results.sort(key=lambda x: x[1],
31                          reverse=True)
32     return tuple([r[0] for r in ranked_results])

```

As bibliotecas *Pandas* e *Sklearn* são importadas para que o código possa acessar suas funcionalidades e realizar a leitura do modelo previamente treinado e armazenado no arquivo *pickle*.

O método `processing_patients` recebe como argumentos um ou mais dados de pacientes, exemplificado no Código 5.17, caso seja fornecido somente um paciente, a função retorna uma lista contendo apenas esse paciente. Caso tenha mais de um argumento, a função cria todas as combinações possíveis de pares de pacientes e compara seus dados usando o modelo previamente carregado. Os resultados são armazenados e retornados em uma lista `results` filtrada para remover duplicatas e classificar os pacientes em ordem decrescente de relevância, com base nos resultados obtidos.

Para aplicar o método criado com precisão científica, é fundamental garantir que os dados fornecidos estejam corretamente normalizados e validados, seguindo as boas práticas de preparação de dados. Abaixo, apresenta-se um exemplo de como os dados devem ser fornecidos ao método para que os resultados obtidos sejam confiáveis e representativos da realidade. Confira o código a seguir:

Listing 5.17: Código de aplicação do modelo

```

1     patient1 = ['James Hughes', 4, 4, 3, 4, 4, 0, 3, 3, 19, 7]

```

```
2 patient2 = ['Christopher Tate', 0, 0, 0, 0, 2, 3, 3, 2, 2, 4]
3 patient3 = ['Kelly Richardson', 2, 1, 2, 3, 2, 1, 3, 0, 10, 5]
4 patient4 = ['Dana Morgan', 0, 1, 2, 2, 0, 1, 3, 1, 5, 4]
5 patient5 = ['Courtney Schultz', 0, 2, 1, 0, 0, 1, 0, 0, 3, 1]
6
7 data = processing_patients(patient1, patient2,
8                             patient3, patient4, patient5)
9
10 for i, item in enumerate(data):
11     print(f"{i+1} - {item[0]}")
```

Para cada paciente, os dados estão no formato: **nome do paciente** seguido pelos valores de 10 atributos  $\alpha_1$  até  $\alpha_6$  (**SOFA**),  $\alpha_7$  (**SPICT**),  $\alpha_8$  (**ECOG**),  $\alpha_9$  (**Total SOFA**) e  $\alpha_{10}$  (**Total AMIB**). Em seguida, ele executa uma função chamada `processing_patients` com os argumentos sendo os dados de todos os cinco pacientes. Essa função retorna uma lista contendo o nome de cada paciente e sua posição na lista.

Na linha nove, a função `Enumerate` é usada para iterar sobre a lista de tuplas enquanto também retorna o índice de cada tupla. A linha 10 é usada para imprimir o número do paciente e seu nome na saída do `loop`.

## 5.4 Resultados e discussões

Esta seção tem como objetivo descrever os resultados mais relevantes obtidos ao longo do processo de estudo, incluindo a apresentação de dados experimentais, análises realizadas do treinamento da Inteligência Artificial (IA) e análises da aplicação de modelos de IA para comparar com dados médicos validados.

Os resultados e discussões apresentados nesta seção contribuem significativamente para a compreensão da área de estudo, possibilitando a elaboração de abordagens mais precisas. Apresenta-se agora o experimento com uma base de dados de 2 milhões de registros gerados:

### Experimento: Análise e treinamento - 2 milhões de casos

Neste experimento, foi realizado um treinamento utilizando uma base de dados restrita, contendo apenas informações dos “paciente 1” e “paciente 2”, a fim de avaliar a capacidade do modelo de inteligência artificial em tomar decisões em cenários limitados e desafiadores.

Neste contexto, cada registro presente na base de dados corresponde às informações de um paciente, contendo um total de 10 variáveis. Para comparar a priorização de pacientes, os dados de um paciente são comparados com os dados de outro paciente, gerando um total de 20 registros na base de dados (com informações de dois pacientes). A última coluna da base de dados representa a decisão considerada correta pelo protocolo AMIB/ABRAMEDE para cada par de pacientes. O valor 1 indica que o paciente 1 deve ter prioridade, o valor 2 indica que o paciente 2 deve ter prioridade, e o valor 0 indica que a decisão deve ser tomada com base no julgamento clínico.

Após a importação dos dados, observou-se que o conjunto de dados é composto por um total de 2.410.364 registros. Destes, 1.206.336 (50,05%) foram classificados como prioritários para o paciente 1, enquanto 1.203.883 (49,95%) foram considerados prioritários para o paciente 2. Ademais, um total de 145 registros (0,01%) a critério do julgamento clínico, seguindo o protocolo AMIB/ABRAMEDE. Para facilitar a compreensão da distribuição desses dados, optou-se pela utilização de um gráfico de pizza, representado na Figura 5.8.

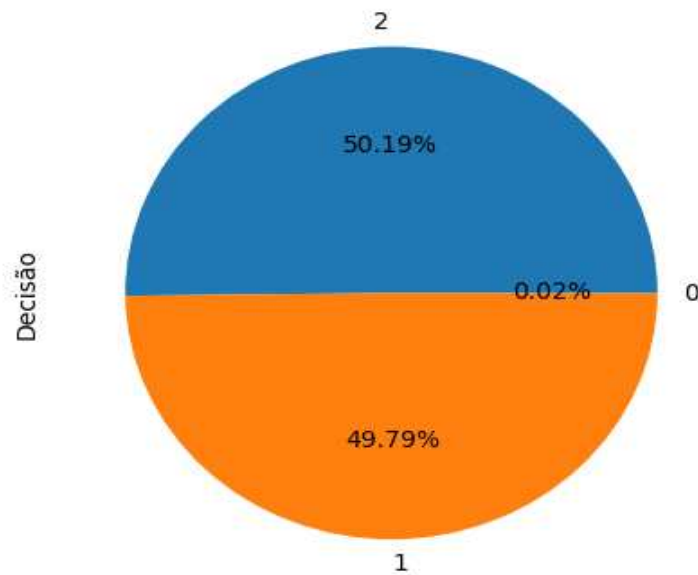


Figura 5.8: Gráfico da distribuição do treinamento dos dados.

Na fase subsequente, os dados foram separados em conjuntos de treinamento e teste, sendo este último composto por 723.110 registros, o que equivale a 30% da amostra total. O conjunto de treinamento, que engloba os demais registros, totaliza 1.687.254 dados. É imprescindível avaliar se tal volume de dados de treinamento é suficiente para alcançar resultados satisfatórios.

Inicia-se com a tabela de importância de cada variável utilizada em um modelo estatístico para predição de resultados sobre o treinamento, onde pode-se perceber que algumas variáveis tem mais preferência no modelo que outras, conforme a Tabela 5.2:

VARIÁVEL	IMPORTÂNCIA
SOFA_total_1	30.454070%
SOFA_total_2	28.957155%
amib_total_1	9.660892%
amib_total_2	9.295515%
icc_1	2.210581%
icc_2	1.882806%
hepat_1	1.558567%
cardi_1	1.543575%
coagu_1	1.519565%
hepat_2	1.491709%
coagu_2	1.457087%
cardi_2	1.452300%
respi_1	1.226547%
neuro_1	1.214867%
neuro_2	1.207092%
respi_2	1.180165%
renal_1	0.949515%
ecog_1	0.945016%
ecog_2	0.900462%
renal_2	0.892513%

**Tabela 5.2:** Resultado da importância das variáveis.

De acordo com esse resultado, as duas variáveis com maior importância no modelo são SOFA\_total\_1, o somatório do SOFA para o paciente 1, e

SOFA\_total\_2, o somatório do SOFA para o paciente 2, representando cerca de 30% e 29% da contribuição, respectivamente. Isso significa que essas variáveis têm um grande impacto na predição dos resultados do modelo.

As variáveis `amib_total_1` e `amib_total_2` também apresentam uma importância relativamente alta, com cerca de 10% cada. As variáveis `icc_1`, `icc_2`, `hepat_1`, `cardi_1` e `coagu_1` têm uma importância entre 2% e 1,5%, enquanto as outras variáveis têm uma importância menor que 1,5%.

A partir desses resultados, pode-se inferir que as variáveis `SOFA_total_1`, `SOFA_total_2`, `amib_total_1` e `amib_total_2` são as mais relevantes para o modelo em questão, enquanto as outras variáveis contribuem com proporções menores para a predição dos resultados.

Essas informações são úteis para entender como o modelo está funcionando e identificar quais variáveis podem ser mais relevantes para uma possível otimização do modelo no futuro.

Para obter uma avaliação mais aprofundada do desempenho do modelo em relação aos dados de treinamento, foram utilizadas diversas métricas, cujos resultados são apresentados na Tabela 5.3:

MÉTRICA	RESULTADO		
Acurácia	99.9999%		
Precisão	100%		
Recall	99.9912%		
F1-Score	99.9955		
AUC-ROC	1.00		
Matriz de Confusão	37	0	0
	0	361525	20
	0	27	361500

**Tabela 5.3:** Resultado das métricas do experimento.

A acurácia do modelo foi de 99,9999%, indicando que ele classificou corretamente a maioria dos dados. A precisão do modelo foi de 100%, o que significa que todas as classificações positivas feitas pelo modelo estavam corretas. O *recall* do modelo RF foi de 99,9912%, indicando que ele classificou corretamente a maioria dos verdadeiros positivos. O F1-Score do modelo foi de 99,9955%, que é uma média harmônica entre precisão e *recall* e é uma medida geral de desempenho do modelo. A curva AUC-ROC teve um valor de

1.00, que é o valor máximo possível e indica que o modelo é capaz de distinguir perfeitamente entre as classes.

Com base na matriz de confusão fornecida, podemos concluir que o modelo de *random forest* apresenta um desempenho bastante satisfatório na classificação dos dados.

### Validação externa

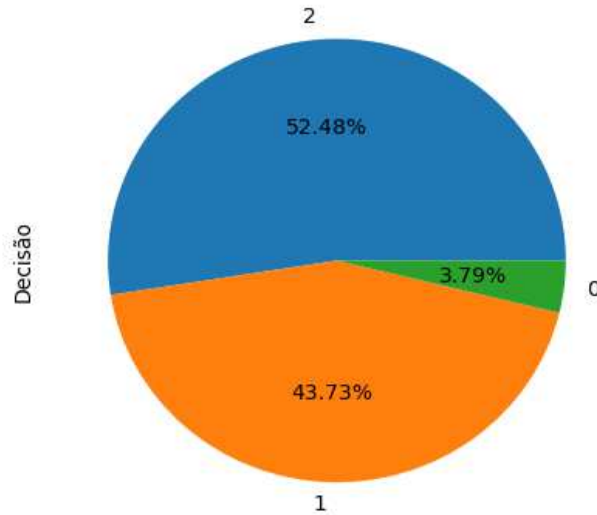
Considerando a alta performance do modelo desenvolvido, foi conduzida uma validação externa para avaliar sua generalização em relação a novos dados. Diferentemente da abordagem tradicional, a qual emprega uma divisão aleatória dos dados em treino e teste, optou-se por importar todos os dados de uma base externa contendo resultados previamente conhecidos e compará-los com as previsões geradas pelo modelo em questão.

O conjunto de dados total da nova base é composto por 1.249.486 registros, distribuídos da seguinte forma: 604.572 registros (48,39%) são destinados à decisão em que o paciente 1 é classificado como prioritário; 595.295 registros (47,64%) são destinados à decisão em que o paciente 2 é classificado como mais prioritário; e 49.619 registros (3,97%) são deixados ao critério do julgamento clínico, conforme o protocolo AMIB/ABRAMEDE. A distribuição dos dados pode ser visualizada por meio do gráfico de pizza na Figura 5.9.

Procedeu-se com a validação do modelo de IA, em que se comparou os resultados obtidos pelo modelo com os resultados previamente registrados na base de dados. A diferença entre as respostas previstas pelo modelo e os valores reais presentes no arquivo foram então calculadas e apresentadas a seguir:

A diferença entre as previsões do modelo e os dados que estão na base nova, é de 50.289 registros, nele estão divididos:

- Quantidade de elementos que estavam com o resultado como zero (0) e o modelo informou como paciente 1, é de 25.970.
- Quantidade de elementos que estavam com o resultado como zero (0) e o modelo informou como paciente 2, é de 23.635.
- Quantidade de elementos que estavam com o resultado como paciente 2 como prioritário e o modelo informou como paciente 1 sendo o prioritário, é de 576.



**Figura 5.9:** Gráfico da distribuição dos dados da validação externa.

- Quantidade de elementos que estavam com o resultado como paciente 1 como prioritário e o modelo informou como paciente 2 sendo o prioritário, é de 108.

Ao ser submetido a uma base de dados desconhecida, o modelo apresentou uma discrepância de 4% (50.289 registros) em relação aos dados de treinamento e 3,97% (49.605 registros) do conjunto total de dados. Observou-se que o modelo fez uma decisão diferente daquela registrada na base de dados para registros em que o valor era zero (0), o que indica a ausência de informação suficiente para determinar a escolha do paciente. Nesses casos, o modelo previu uma prioridade aos pacientes 1 ou 2, o que representa uma variação em relação às decisões da base de dados original.

Considerados “erros graves” são os 684 registros que correspondem a 0,03% da base de testes externos. Essa porcentagem se refere aos casos em que o modelo previu o paciente 2 como prioritário, quando na verdade o resultado gerado era o paciente 1, ou vice-versa, de acordo com o protocolo AMIB/ABRAMEDE. Totalizando uma acurácia de 99.7% na base de testes externos.

## **Avaliando a eficácia da aplicação de modelo em dados validados por médicos**

Para se ter uma amostra de como o modelo pode ser utilizado em um hospital como sendo um auxiliador, propõem-se estudar a eficiência da listagem realizada por médicos comparado com a listagem que o modelo realizou.

No início deste estudo os médicos voluntários acessaram o sistema de validação 5.2, onde realizaram uma análise de 10 pacientes que estavam como julgamento clínico, gerado via algoritmo baseado no protocolo AMIB/ABRAMEDE. Esses dados podem ser aplicados para uma validação do modelo criado neste estudo, com o objetivo de entender a porcentagem de acurácia nessa lista.

No primeiro exemplo, o médico Matheus validou cuidadosamente a listagem dos pacientes com seus dados conforme o protocolo e levantou sugestões para melhorias com base na experiência em leitos de UTI. No entanto, para o presente estudo, a alocação de leitos seguirá rigorosamente o protocolo AMIB/ABRAMEDE. A listagem fornecida pelo médico é apresentada na parte esquerda da Tabela 5.4.

Para a validação do modelo proposto, foi realizada a aplicação do Código 5.17 utilizando-se os mesmos registros que o médico validou inicialmente. A listagem resultante é apresentada na parte direita da Tabela 5.4.

A tabela apresenta dois conjuntos de resultados: na posição esquerda da tabela, corresponde à validação médica, onde os nomes foram classificados manualmente por um médico experiente; na posição direita da tabela conjunto corresponde à validação da IA, onde os nomes foram classificados automaticamente pelo modelo de aprendizado de máquina desenvolvido neste estudo.

Ao comparar os dois conjuntos de resultados, podemos observar que os cinco primeiros nomes coincidem em ambos os conjuntos, indicando que tanto o método manual quanto o automático foram capazes de identificar corretamente esses nomes. Pode-se perceber na posição 8, a qual ambos decidiram a listagem como Jennifer Roberts, sendo dessa forma como uma validação correta. Entretanto, as diferenças começam a surgir a partir do sexto nome. Enquanto a validação do médico Matheus classificou o nome como “Jason Brown”, a validação da IA classificou-o como “Tommy

Validação Médica (Matheus)	Validação da IA (Matheus)
1 - Kelly Richardson	1 - Kelly Richardson
2 - Kelly Hall	2 - Kelly Hall
3 - Caitlin Wong	3 - Caitlin Wong
4 - James Mcconnell	4 - James Mcconnell
5 - Evan Griffin Jr.	5 - Evan Griffin Jr.
6 - Jason Brown	6 - Tommy Hicks
7 - Cynthia Lane	7 - Brandon Spencer
8 - Jennifer Roberts	8 - Jennifer Roberts
9 - Brandon Spencer	9 - Cynthia Lane
10 - Tommy Hicks	10 - Jason Brown

**Tabela 5.4:** Listagem comparativa entre médico e modelo.

Hicks”. Similarmente, a validação médica classificou o nome como “Cynthia Lane”, enquanto a validação da IA classificou-o como “Brandon Spencer”, totalizando 40% de erro.

Realizando uma contagem de acerto dessas listas, o modelo de IA acertou 6 colocações iguais ao que o médico havia listado, isso pode resultar em 60% de acerto em uma listagem pequena e imprecisa, pois esses dados o protocolo AMIB indica Julgamento Clínico.

Essas diferenças não necessariamente indicam que a IA esteja errada, pois a decisão médica é muito complexa. É importante ressaltar que a validação da IA é um processo automatizado, e, é possível que a IA possa ser aprimorada por meio de ajustes em seu algoritmo e treinamento com um conjunto de dados diferente, melhor validado.

No segundo exemplo, a médica Isadora validou rigorosamente a listagem dos pacientes com seus dados conforme o protocolo, comprovando-o por meio de uma validação que levou uma hora e trinta minutos para ser realizada. Durante a validação, a médica na forma de amostragem dos dados, e melhorias no protocolo, com base na sua experiência, as mudanças adicionais são nas variáveis que o modelo analisa e contexto do paciente. No entanto, para o presente estudo, será mantida a alocação de leitos conforme o protocolo AMIB/ABRAMEDE. A listagem fornecida pelo médico é apresentada na parte esquerda da Tabela 5.5. O modelo foi aplicado utilizando o Código 5.17, com os mesmos registros da médica de forma aleatória, e a listagem resultante é apresentada na parte direita da Tabela 5.5.

Validação Médica (Isadora)	Validação da IA (Isadora)
1 - Christopher McCormick	1 - Christopher McCormick
2 - Jackie Spence	2 - Jackie Spence
3 - Curtis Anderson	3 - Curtis Anderson
4 - Thomas Ross	4 - Thomas Ross
5 - Edward Larsen	5 - Holly Atkinson
6 - Elijah Diaz	6 - Edward Larsen
7 - Dennis Waller	7 - Mia Andrade
8 - Mia Andrade	8 - Jimmy Reed
9 - Jimmy Reed	9 - Elijah Diaz
10 - Holly Atkinson	10 - Dennis Waller

**Tabela 5.5:** Lista comparativa - médica vs modelo.

Ao se comparar as duas listas, nota-se uma concordância em quatro dos cinco primeiros pacientes identificados, entretanto, após a quinta posição, houve uma inversão do modelo de Inteligência Artificial em relação à posição de Holly Atkinson, que foi identificada como a décima posição pela validação médica. Em relação à posição seis, Edward Larsen, o modelo obteve uma proximidade com a posição cinco, que foi validada pela médica. As posições seguintes apresentaram inversões em relação à validação médica.

Realizando uma contagem de acerto dessas listas, o modelo de IA acertou 4 colocações iguais ao que o médico havia listado, e uma colocação muito próxima, calcula-se como 5%, resultando em 45% de acerto em uma listagem pequena e imprecisa, pois esses dados o protocolo AMIB/ABRAMEDE indica julgamento clínico.

Deve-se ressaltar que a IA utilizada neste estudo apresentou um alto nível de acerto na identificação dos pacientes, e que as diferenças observadas em relação à validação médica podem ser explicadas por variações na interpretação dos registros médicos e nas técnicas de análise utilizadas.

É importante considerar que a utilização de tecnologias baseadas em IA pode ser uma ferramenta útil no processo de identificação de pacientes com determinadas condições médicas, mas que a validação médica ainda é essencial para garantir a qualidade dos resultados.

Durante uma reunião com a médica, foram apresentados os resultados da pesquisa e solicitada uma avaliação crítica de sua perspectiva em relação ao modelo proposto. A seguir, são apresentados alguns dados dos pacientes:

- Em relação às comparações da posição 5, o modelo previu a Holly Atkinson: “O sistema cardiovascular costuma ser prioritário. O uso de vasopressor, associado à plaquetopenia, garante a Holly maior prioridade que os demais.”
- Em relação às comparações da posição 6, o modelo previu para Edward Larsen o que a médica havia validado como posição 5: “Edward ainda não está com tratamento otimizado e apresenta sinais de deterioração caso não receba tratamento.”
- Em relação às comparações da posição 7, Mia Andrade, pela posição do modelo computacional: “Mia inspira cuidados. Caso não responda inicialmente ao tratamento, terá alta mortalidade devido à combinação de plaquetopenia grave e IR, que evoluirá para indicação HD. Essa deterioração aparece em Elijah - que já tem indicação de HD e sinais de alteração cognitiva. É uma escolha difícil entre Mia e Elijah, mas considerando que Mia tem maiores chances de sobrevivência, opta-se por ela.”
- Em relação às comparações da posição 9: “É difícil definir quem ‘ganha’ entre Jimmy e Elijah, mas considerando que Jimmy tem maior chance de sobrevivência, escolhe-se ele.”
- Em relação às comparações da posição 10: “Dennis tem um alto risco devido à associação de plaquetopenia e indicação de HD. É por isso que antes dei preferência a ele, mas agora reavaliando, acredito que de todos ele tem as maiores condições de se sair bem sem um leito de UTI, por isso, opta-se por deixá-lo por último.”

## Descoberta de possível divergência: investigando uma hipótese

Durante a realização de diversos experimentos com o modelo de Inteligência Artificial, surgiu a seguinte questão: “Considerando que o algoritmo *random forest* realiza a divisão das variáveis de maneira individual no algoritmo Árvore de Decisão, é possível que o algoritmo esteja aprendendo de forma equivocada e isolada?”

Para solucionar esse possível problema, buscou-se uma abordagem que unisse matematicamente as variáveis dos pacientes em comparação, considerando que os valores em cada coluna poderiam ser manipulados matematicamente a fim de obter uma união entre os dados.

502_vs_502,0,16,0,16,0,16,0,1,0,16,0,1,0,0,0,0,289,0,16,0
502_vs_4076,3,4,1,12,4,0,-3,4,0,16,-3,4,-3,0,0,0,5,204,-2,24,2
502_vs_6140,0,16,3,4,1,12,1,0,1,12,-1,2,-3,0,0,0,6,187,-1,20,1
502_vs_6141,4,0,1,12,3,4,-1,2,0,16,0,1,-3,0,0,0,7,170,-1,20,1
502_vs_6142,2,8,0,16,4,0,0,1,3,4,-2,3,0,0,0,0,9,136,3,4,2
502_vs_6409,4,0,4,0,4,0,1,0,4,0,-1,2,-3,0,0,0,17,0,0,16,1
502_vs_6964,4,0,4,0,4,0,1,0,4,0,0,1,0,0,0,0,17,0,3,4,1
502_vs_7154,0,16,1,12,0,16,-1,2,2,8,-3,4,0,0,0,0,2,255,0,16,2
502_vs_7420,0,16,1,12,0,16,-1,2,0,16,-3,4,-3,0,0,0,0,289,-3,28,1
502_vs_7722,0,16,0,16,0,16,-3,4,0,16,1,0,0,0,0,0,-3,340,0,16,1
502_vs_7850,0,16,0,16,0,16,-3,4,0,16,-1,2,0,0,0,0,-3,340,0,16,1
502_vs_7929,3,4,4,0,4,0,1,0,4,0,1,0,0,0,0,0,16,17,3,4,2

**Figura 5.10:** Estrutura de dados montada com a diferença e produto.

Para cada variável dos pacientes, foi realizado o cálculo da diferença e do produto entre os valores dos pacientes, gerando assim duas novas variáveis que representam a união dos dados dos pacientes. Por exemplo, se o Paciente 1 e o Paciente 2 possuem a mesma variável Neurológico (`neuro_1` e `neuro_2`, respectivamente), a operação de subtração foi realizada para gerar a nova variável `dif_neuro` e a operação de multiplicação para gerar a nova variável `prod_neuro`. Observa-se na Figura 5.10 que os dados dos pacientes não são apresentados individualmente, mas sim como dados unidos que oferecem um número para ser treinado. Essa abordagem permitiu que as variáveis fossem tratadas de forma conjunta, garantindo que cada variável estivesse com o mesmo valor de propriedade nos dados do paciente.

Os dados de todos os pacientes foram submetidos ao mesmo processo, considerando todas as colunas disponíveis. O código utilizado encontra-se descrito a seguir:

```

1     def reformat(row):
2         return (
3             row[0],
4             int(row[1]) - int(row[11]),
5             int(row[1]) * int(row[11]),
6             int(row[2]) - int(row[12]),
7             int(row[2]) * int(row[12]),
8             int(row[3]) - int(row[13]),
9             int(row[3]) * int(row[13]),

```

```
10         int (row[4]) - int (row[14]),
11         int (row[4]) * int (row[14]),
12         int (row[5]) - int (row[15]),
13         int (row[5]) * int (row[15]),
14         int (row[6]) - int (row[16]),
15         int (row[6]) * int (row[16]),
16         int (row[7]) - int (row[17]),
17         int (row[7]) * int (row[17]),
18         int (row[8]) - int (row[18]),
19         int (row[8]) * int (row[18]),
20         int (row[9]) - int (row[19]),
21         int (row[9]) * int (row[19]),
22         int (row[10]) - int (row[20]),
23         int (row[10]) * int (row[20]),
24         row[21]
25     )
26
27     def main():
28         reader =
29             csv.reader(open(f' /Database_training_7b1.csv',
30                             'r'), csv.excel)
31         writer =
32             csv.writer(open(f'Arquivos_Tot/refor_7b1.csv',
33                             'w'))
34         for row in reader:
35             if not row:
36                 continue
37             writer.writerow(reformat(row))
```

O resultado do algoritmo gera um novo arquivo com dados suficiente para um novo treinamento. Após rodar todas as métricas, conclui-se que os resultados de treinamento entre variáveis dos pacientes e as variáveis unidas calculadas, tem uma diferença insignificante no resultado final, sendo que no treinamento com os dados esclarecidos no protocolo AMIB/ABRAMEDE a validação externa obteve dados semelhantes em relação ao treinamento.

Portanto a hipótese levantada de que as variáveis individualizadas de cada paciente utilizadas no treinamento, não tem uma significativa diferença no treinamento, da variáveis unidas matematicamente, produto e diferença dos pacientes.

## 5.5 Dificuldades

Após a implementação do software para a primeira validação com os médicos voluntários, ocorreu uma perda de contato com o médico de apoio, além de eventos nacionais e locais que interferiram na validação planejada. Inicialmente, havia a expectativa e o planejamento de contar com uma equipe de 10 médicos validando a ordenação de 10 grupos de 10 pacientes cada, totalizando 100 grupos de pacientes para realizar o treinamento do modelo. No entanto, somente três médicos voluntários validaram um grupo cada, resultando em apenas 28 registros, dos quais dois foram considerados inválidos.

Após a aplicação das regras do protocolo AMIB/ABRAMEDE para a geração de possibilidades de pacientes, foram obtidos um total de 72.500 mil registros. A comparação binária desses registros resultou em um arquivo de tamanho significativo de 291 GB, contendo mais de 6 milhões de registros. A tentativa de processar todos os dados usando um único computador local e, posteriormente, a abordagem de computação em nuvem foram insuficientes para treinar o modelo com sucesso.

## 5.6 Resumo do capítulo

Neste capítulo, foi apresentado a implementação do sistema de validação dos médicos voluntários, o gerenciamento dos dados aplicados ao modelo computacional, bem como a leitura e tratamento de dados, juntamente com a análise exploratória. Também foram descritos o treinamento e a validação do modelo, bem como os resultados e discussões do experimento realizado com 2 milhões de casos e dificuldades encontradas.



---

# Capítulo 6

## Considerações Finais

---

*If you can look into the seeds of time,  
And say which grain will grow and which will not;  
Speak then to me.*

*William Shakespeare, British author (1564–1616)*

### 6.1 Conclusão

**N**este trabalho, foi proposto um modelo computacional baseado em inteligência artificial para auxiliar os médicos na avaliação da gravidade dos pacientes e na tomada de decisões sobre a priorização do acesso aos leitos de UTI, considerando a sua escassez.

Por meio da integração entre algoritmos de inteligência artificial com modelagem matemática, foi desenvolvido um modelo baseado em um protocolo de triagem validado pela Associação de Medicina Intensiva Brasileira (AMIB) e Associação Brasileira de Medicina Diagnóstica (ABRAMEDE). O uso desse modelo resultou em uma indicação de pacientes sugerida de forma mais transparente e imparcial, permitindo um auxílio ao médico mais confiável e eficaz sobre os pacientes.

A pesquisa foi conduzida utilizando o método *Design Science Research* com o objetivo de compreender o assunto em questão e propor soluções efetivas. Inicialmente, foi realizada uma revisão sistemática da literatura científica disponível, a fim de fundamentar o conhecimento, comportamento e aplicabilidade do protocolo AMIB/ABRAMEDE, apresentando uma análise detalhada. Em seguida, a pesquisa evoluiu para a construção e avaliação de

um artefato em um ambiente controlado, utilizando um software com o modelo e a estrutura necessários para proceder com a homologação. A revisão possibilitou a compreensão completa dos algoritmos utilizados e seus respectivos fundamentos, oferecendo um embasamento sólido para a construção do artefato.

A fase de geração de dados foi essencial para identificar os critérios e regras necessários para a construção de uma base de dados com pacientes hipotéticos. Cada etapa deste processo exigiu soluções específicas e abordagens distintas. Durante a última fase da geração de dados, foram empregadas técnicas de transitividade e antissimétrica para obter resultados mais eficientes e confiáveis. Quando o protocolo exigia julgamento clínico, um *software* de fácil utilização foi implementado para permitir que médicos voluntários validassem os dados selecionados.

Embora a validação empírica tenha sido realizada com participantes voluntários, os resultados obtidos não atenderam às expectativas pré-estabelecidas, o que evidenciou uma limitação na amplitude da amostra. É importante destacar que esta fase de geração de dados é crítica e altamente complexa no desenvolvimento do trabalho.

A utilização do algoritmo supervisionado proposto, por meio de técnicas de inteligência artificial, resultou em uma alta complexidade computacional. Devido à falta de recursos financeiros para investimentos em infraestrutura, o protótipo enfrentou dificuldades em processar grandes volumes de dados. Como resultado, o treinamento do modelo foi limitado a uma quantidade menor de possibilidades.

Durante a fase de validação do modelo proposto neste estudo, conduzida por médicos voluntários, observou-se que as regras e variáveis presentes no protocolo AMIB/ABRAMEDE não foram consideradas suficientes para garantir completa precisão e a aplicabilidade do modelo na prática clínica. Com base nisso, os médicos sugeriram a inclusão de outras variáveis, tais como idade, tempo de permanência no hospital e doença de origem, com o objetivo de contextualizar adequadamente os dados dos pacientes e, assim, aprimorar a precisão e eficácia do modelo em cenários reais de atendimento médico.

Com base na acurácia de 99.99% nos resultados obtidos nos dados de treinamento, e uma acurácia de 99.7% de acurácia para dados externos, podemos inferir que o uso do modelo proposto de inteligência artificial baseado no protocolo da AMIB/ABRAMEDE de 2020 é uma alternativa viável e eficiente para auxiliar os médicos intensivistas a tomarem decisões mais rapidamente,

gerenciarem leitos e aumentarem a transparência em Unidades de Terapia Intensiva, desde que variáveis adicionais sugeridas sejam incluídas. Além disso, o modelo desenvolvido pode ser adaptado para outras áreas que requerem tomadas de decisão, como indicado na Seção 6.2. Esta conclusão é suportada pelos dados gerados e analisados neste estudo.

## 6.2 Trabalhos futuros

Recomenda-se considerar uma validação mais ampla por equipes médicas dedicadas, e remuneradas que possam avaliar e aprimorar o modelo proposto por meio de diferentes técnicas de validação, tais como a expansão de validação por dados de pacientes reais e a validação por especialistas da área, a fim de garantir a confiabilidade e a aplicabilidade do modelo em cenários práticos.

Recomenda-se que a expansão deste estudo inclua casos de paraplégicos ou outras situações atípicas, a fim de avaliar a aplicabilidade e a precisão do modelo em uma ampla gama de pacientes e condições clínicas. Isso permitiria uma maior generalização dos resultados obtidos e uma validação mais robusta do modelo proposto.

Considerando trabalhos futuros, sugere-se explorar a possibilidade de adaptar o protocolo AMIB/ABRAMEDE para incorporar as particularidades de cada hospital e processo de UTI, visando aprimorar a eficácia e precisão do modelo proposto em cenários práticos de atendimento médico.

Sugere-se aprimorar a eficiência dos processos mediante a utilização de uma máquina com hardware mais robusto, tais como um processador de maior capacidade, maior quantidade de memória RAM e um disco rígido com maior capacidade de armazenamento, de forma a possibilitar a execução mais ágil das tarefas e o processamento de grandes conjuntos de dados. Além disso, essa atualização permitirá a aplicação de outros algoritmos complexos, contribuindo para aumentar a precisão dos resultados obtidos.



---

## Bibliografia

---

- [1] C. Baia. *Decision tree e random forest*. <http://carlosbaia.com/2016/12/24/decision-tree-e-random-forest/>, 2016. Acessado Março 2022.
- [2] M. L. Bastos e A. J. d. C. C. Lins. *Chronic illness diagnosis helper: proposta de uma ferramenta para auxílio ao diagnóstico de doenças crônicas através da análise histórica de relatos sintomáticos*. *Revista de Engenharia e Pesquisa Aplicada*, 5(2):51–61, 2020.
- [3] Z. Bernard. *So, what is machine learning anyways? here's a quick breakdown*. <https://www.businessinsider.com/what-is-machine-learning-quick-explainer-2017-11>, nov 2017.
- [4] E. L. D. Biddison, R. Faden, H. S. Gwon, D. P. Mareiniss, A. C. Regenberg, M. Schoch-Spana, J. Schwartz e E. S. Toner. *Too many patients... a framework to guide statewide allocation of scarce mechanical ventilation during disasters*. *Chest*, 155(4):848–854, 2019.
- [5] C. M. Bishop e N. M. Nasrabadi. *Pattern recognition and machine learning*, volume 4. Springer, 2006.
- [6] L. Breiman. *Random forests*. *Machine learning*, 45(1):5–32, 2001.
- [7] P. Bruce, A. Bruce e P. Gedeck. *Practical statistics for data scientists: 50+ essential concepts using r and python*. O'Reilly Media, 2020.
- [8] L. Brunialti, S. Peres, V. Freire e C. Lima. *Aprendizado de maquina em sistemas de recomendacao baseados em conteudo textual: Uma revisao sistematica*. Em *Anais do XI Simpósio Brasileiro de Sistemas de Informação*, páginas 203–210. SBC, 2015.
- [9] T. P. Cappello. *Alocação de leitos em unidade de terapia intensiva na pandemia covid-19: critérios de priorização*. Tese Doutoral, Universidade de São Paulo, 2022.

- [10] D. Chappell. *Introducing azure machine learning. A guide for technical professionals, sponsored by microsoft corporation*, 2015.
- [11] E. Charniak. *Introduction to artificial intelligence*. Pearson Education India, 1985.
- [12] D. Cielen e A. Meysman. *Introducing data science: big data, machine learning, and more, using python tools*. Simon and Schuster, 2016.
- [13] Conselho Federal de Medicina (CFM). *Pandemia aumenta em 45% número de leitos de uti, mas distribuição ainda é marcada pela desigualdade*, agosto 2020. URL: <https://portal.cfm.org.br/noticias/>.
- [14] A. Q. Côrtes, J. P. d. O. Silva, T. F. Rebouças et al.. *A prioridade de leitos de uti em época de pandemia: Uma possível adequação do critério de idade proposto pelo protocolo de alocação de recursos em esgotamento durante a pandemia da covid-19 criado pela abramede e amib à luz do princípio da dignidade da pessoa humana. SEMOC-Semana de Mobilização Científica-Envelhecimento em tempos de pandemias*, 2020.
- [15] C. M. A. Costa, F. C. L. Pompermayer e A. da Silva Costa. *Protocolos em tempos de pandemia por covid-19 com critérios para alocação de recursos escassos: podem considerá-los (in) justos? DIVERSITATES International Journal*, 12(2):131–162, 2020.
- [16] F. G. Cozman. *O futuro da (pesquisa em) inteligência artificial: algumas direções. Revista USP*, (124):11–20, 2020.
- [17] A. Cutler, D. R. Cutler e J. R. Stevens. *Random forests. Ensemble machine learning: Methods and applications*, páginas 157–175, 2012.
- [18] P. F. da Silva. *Avanços, retrocessos e práticas em pesquisa*. 2022.
- [19] D. Dall’Agnol. *Cilcos de vida ou performance?* 2021.
- [20] T. G. Dietterich. *Ensemble methods in machine learning*. Em *Multiple Classifier Systems: First International Workshop, MCS 2000 Cagliari, Italy, June 21–23, 2000 Proceedings 1*, páginas 1–15. Springer, 2000.
- [21] C. Feng, L. Wang, X. Chen, Y. Zhai, F. Zhu, H. Chen, Y. Wang, X. Su, S. Huang, L. Tian et al.. *A novel artificial intelligence-assisted triage tool to aid in the diagnosis of suspected covid-19 pneumonia cases in fever clinics. Annals of Translational Medicine*, 9(3), 2021.

- [22] Y. Freund e R. E. Schapire. *A decision-theoretic generalization of on-line learning and an application to boosting*. *Journal of computer and system sciences*, 55(1):119–139, 1997.
- [23] A. Géron. *Hands-on machine learning with scikit-learn, keras, and tensorflow*. "O'Reilly Media, Inc.", 2022.
- [24] J. Hao e T. K. Ho. *Machine learning made easy: a review of scikit-learn package in python programming language*. *Journal of Educational and Behavioral Statistics*, 44(3):348–361, 2019.
- [25] T. Hastie, R. Tibshirani, J. H. Friedman e J. H. Friedman. *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer, 2009.
- [26] A. R. Hevner, S. T. March, J. Park e S. Ram. *Design science in information systems research*. *misquarterly. MISQ Discovery*, 28(1), 2004.
- [27] J. L. Izquierdo, J. Ancochea, J. B. Soriano, S. C.-. R. Group et al.. *Clinical characteristics and prognostic factors for intensive care unit admission of patients with covid-19: retrospective study using machine learning and natural language processing*. *Journal of medical Internet research*, 22(10):e21801, 2020.
- [28] J. D. Kelleher, B. Mac Namee e A. D'arcy. *Fundamentals of machine learning for predictive data analytics: algorithms, worked examples, and case studies*. MIT press, 2020.
- [29] M. Kogien et al.. *Implantação da ferramenta kanban e da estratégia just-in-time adaptados para a gestão do tempo de permanência do paciente em uma unidade de terapia intensiva*. 2017.
- [30] O. Kramer e O. Kramer. *Scikit-learn. Machine learning for evolution strategies*, páginas 45–53, 2016.
- [31] L. Kretzer, E. Berbigier, R. Lisboa, A. C. Grumann e J. Andrade. *Protocolo amib de alocação de recursos em esgotamento durante a pandemia por covid-19*. *Associação de Medicina Intensiva Brasileira*. [Versão eletrônica]. Obtido em, 28, 2020.
- [32] R. Kurzweil, R. Richter, R. Kurzweil e M. L. Schneider. *The age of intelligent machines*, volume 580. MIT press Cambridge, 1990.

- [33] S. Lambden, P. F. Laterre, M. M. Levy e B. Francois. *The sofa score—development, utility and challenges of accurate assessment in clinical trials*. *Critical Care*, 23(1):1–9, 2019.
- [34] Y. Liu, Y. Wang e J. Zhang. *New machine learning algorithm: Random forest*. Em *Information Computing and Applications: Third International Conference, ICICA 2012, Chengde, China, September 14-16, 2012. Proceedings 3*, páginas 246–252. Springer, 2012.
- [35] L. C. Lobo. *Inteligência artificial e medicina*. *Revista Brasileira de Educação Médica*, 41:185–193, 2017.
- [36] A. C. Lorena e A. C. De Carvalho. *Uma introdução às support vector machines*. *Revista de Informática Teórica e Aplicada*, 14(2):43–67, 2007.
- [37] P. A. Medeiros, E. Landulfo, P. I. d. B. DE Iniciação, S. A. PIBIC, S. A. PROBIC, S. A. PIBITI et al.. *Metodologia de tratamento de dados lidar com python e estruturação da obtenção de dados*. 2022.
- [38] W. Medri. *Análise exploratória de dados*. Londrina: Universidade Estadual de Londrina, 2011.
- [39] M. Meneghetti e M. Meneghetti. *Python mini-tutorial. Introduction to Gravitational Lensing: With Python Examples*, páginas 371–380, 2021.
- [40] N. Middleton e R. Schneeman. *Heroku: up and running: effortless application deployment and scaling*. "O'Reilly Media, Inc.", 2013.
- [41] B. Momjian. *Postgresql: introduction and concepts*, volume 192. Addison-Wesley New York, 2001.
- [42] I. L. Mota e R. Queiroz. *Distúrbios do equilíbrio ácido básico e gasometria arterial: uma revisão crítica*. *Revista Digital*, Buenos Aires, 14, 2010.
- [43] P. Motoryn. [9 em cada 10 médicos tiveram saúde mental afetada na pandemia, diz pesquisa](https://www.poder360.com.br/coronavirus/9-em-cada-10-medicos-tiveram-saude-mental-afetada-na-pandemia-diz-pesquisa/), apr 2021. URL: <https://www.poder360.com.br/coronavirus/9-em-cada-10-medicos-tiveram-saude-mental-afetada-na-pandemia-diz-pesquisa/>.
- [44] K. P. Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

- [45] S. Murthy, A. Leligdowicz e N. K. Adhikari. *Intensive care unit capacity in low-income countries: a systematic review*. *PloS one*, 10(1): e0116949, 2015.
- [46] M. d. A. Nakashima et al.. *Intervindo na via do óxido nítrico durante a sepse experimental: papel dos inibidores da fosfodiesterase 5 sobre as disfunções cardiovasculares induzidas pela sepse*. 2020.
- [47] D. D. Palmer. *Tokenisation and sentence*. *Handbook of natural language processing*, página 11, 2000.
- [48] K. Peffers, T. Tuunanen, M. A. Rothenberger e S. Chatterjee. *A design science research methodology for information systems research*. *Journal of management information systems*, 24(3):45–77, 2007.
- [49] M. Pimentel, D. Filippo e T. M. Santos. *Design science research: pesquisa científica atrelada ao design de artefatos*. *RE@ D-Revista de Educação a Distância e Elearning*, 3(1):37–61, 2020.
- [50] H. A. Pinto, W. McCaskill-Stevens, P. Wolfe e A. C. Marcus. *Physician perspectives on increasing minorities in cancer clinical trials: an eastern cooperative oncology group (ecog) initiative*. *Annals of Epidemiology*, 10(8):S78–S84, 2000.
- [51] J. R. Quinlan. *Induction of decision trees*. *Machine learning*, 1:81–106, 1986.
- [52] J. G. R. Ramos. *Avaliação de um instrumento de auxílio à tomada de decisão para a priorização de vagas em unidades de terapia intensiva*. Tese Doutoral, Universidade de São Paulo, 2018.
- [53] D. D. Rodrigues. *Design science research como caminho metodológico para disciplinas e projetos de design da informação | design science research as methodological path for information design subjects and projects*. *InfoDesign-Revista Brasileira de Design da Informação*, 15(1): 111–124, 2018.
- [54] L. B. Rodriguez, L. B. Rodriguez, M. S. M. C. Leão, M. J. G. d. Mello, Z. d. R. Cavalcanti e A. V. X. d. M. Madureira. *Pergunta surpresa e spict (supportive and palliative care indicators tool): existe concordância entre os médicos?* 2021.
- [55] S. J. Russell. *Artificial intelligence a modern approach*. Pearson Education, Inc., 2010.

- [56] J. F. SÁ. *Mineração de dados usando o algoritmo chaid*. 2012.
- [57] A. Sassi. *Código fonte do trabalho*. <https://github.com/alisonsassi95/MasterMedicalDecisionSupport>, apr 2023. Código fonte para consulta e replicação deste trabalho.
- [58] T. C. Silva e L. Zhao. *Machine learning in complex networks*, volume 1. Springer, 2016.
- [59] H. A. Simon. *The sciences of the artificial, reissue of the third edition with a new introduction by john laird*. MIT press, 2019.
- [60] A. Smola e S. Vishwanathan. *Introduction to machine learning*. Cambridge University, UK, 32(34):2008, 2008.
- [61] A. M. d. Souza. *Machine learning e a evasão escolar: análise preditiva no suporte à tomada de decisão*. Tese Doutoral, Mestrado em Sistemas de Informação e Gestão do Conhecimento, 2020.
- [62] D. Souza-Couto, N. B. da Silva e E. J. R. Cardoso. *Avaliação do conhecimento de estudantes da área da saúde sobre a escala de coma de glasgow em uma universidade de minas gerais*. *Research, Society and Development*, 10(9):e2410917798–e2410917798, 2021.
- [63] M. T. A. Steiner, N. Y. Soma, T. Shimizu, J. C. Nievola, F. LOPES e A. Smiderle. *Data-mining como suporte à tomada de decisões-uma aplicação no diagnóstico médico*. *XXXVI Simpósio Brasileiro de Pesquisa Operacional, "O impacto da pesquisa operacional nas novas tendências multidisciplinares*, 23:96–107, 2004.
- [64] W. M. Tingey e K. Rose. *Foliar insecticides for potato insect control, freeville, new york, 1979*. *Insecticide and Acaricide Tests*, 5(1):101–101, 1980.
- [65] I. H. Witten, E. Frank, M. A. Hall, C. J. Pal e M. DATA. *Practical machine learning tools and techniques*. Em *DATA MINING*, volume 2, página 4, 2005.
- [66] Z.-H. Zhou. *Ensemble methods: foundations and algorithms*. CRC press, 2012.

*This document was typeset on May 15, 2023 using class `RG-BOK`  $\alpha$ 2.14 for L<sup>A</sup>T<sub>E</sub>X<sub>2</sub> $\epsilon$ . As of the time of writing this document, this class is not publicly available. Only members of *The Distributed Group (TDG)* and the *Applied Computing Research Group (ACR)* are allowed to typeset their documents using this class.*