FERNANDA DE ANDRADE FLOR

Modelos de Inteligência Artificial para Predição de Indicação e do Risco de Lesões em Atletas de Futebol Feminino

UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE MATEMÁTICA
2026

FERNANDA DE ANDRADE FLOR

Modelos de Inteligência Artificial para Predição de Indicação e do Risco de Lesões em Atletas de Futebol Feminino

Dissertação apresentada ao Programa de Pós-Graduação em Matemática da Universidade Federal de Uberlândia, como parte dos requisitos para obtenção do título de **MESTRE EM MATEMÁTICA.**

Área de Concentração: Matemática.
Linha de Pesquisa: Matemática Aplicada.

Orientadora: Profa. Dra. Rosana Sueli da Motta Jafelice.
Corientador: Prof. Dr. José Waldemar da Silva.

UBERLÂNDIA - MG
2026

Ficha Catalográfica Online do Sistema de Bibliotecas da UFU
com dados informados pelo(a) próprio(a) autor(a).

F632 2026

Flor, Fernanda de Andrade, 2000-
Modelos de Inteligência Artificial para Predição de Indicação e do Risco de Lesões em Atletas de Futebol Feminino [recurso eletrônico] / Fernanda de Andrade Flor. - 2026.

Orientadora: Rosana Sueli da Motta Jafelice.
Coorientador: José Waldemar da Silva.
Dissertação (Mestrado) - Universidade Federal de Uberlândia, Pós-graduação em Matemática.
Modo de acesso: Internet.
DOI http://doi.org/10.14393/ufu.di.2026.156
Inclui bibliografia.
Inclui ilustrações.

1. Matemática. I. Jafelice, Rosana Sueli da Motta ,1964-, (Orient.). II. Silva, José Waldemar da,1975-, (Coorient.). III. Universidade Federal de Uberlândia. Pós-graduação em Matemática. IV. Título.

CDU: 51

Bibliotecários responsáveis pela estrutura de acordo com o AACR2:
Gizele Cristine Nunes do Couto - CRB6/2091
Nelson Marcos Ferreira - CRB6/3074

UNIVERSIDADE FEDERAL DE UBERLÂNDIA
Coordenação do Programa de Pós-Graduação em Matemática
Av. João Naves de Ávila, 2121, Bloco 1F, Sala 1F 158 - Bairro Santa Mônica, Uberlândia-MG, CEP 38400-902
Telefone: (34) 3239-4209/4154 - www.ppmat.ime.ufu.br - pgmat@ime.ufu.br

ATA DE DEFESA - PÓS-GRADUAÇÃO

Programa de Pós-Graduação em:	Matemática				
Defesa de:	Dissertação de Mestrado Acadêmico, 131, PPGMAT				
Data:	25/02/2026	Hora de início:	14:00	Hora de encerramento:	16:00
Matrícula do Discente:	12412MAT002				
Nome do Discente:	Fernanda de Andrade Flor				
Título do Trabalho:	Modelos de Inteligência Artificial para Predição de Indicação e do Risco de Lesões em Atletas de Futebol Feminino				
Área de concentração:	Matemática				
Linha de pesquisa:	Matemática Aplicada				
Projeto de Pesquisa de vinculação:	Estudo das Incertezas em Modelos Matemáticos via Conjuntos Fuzzy do Tipo 1 e do Tipo 2 Intervalar				

Reuniu-se na Sala 1F 119 (Sala Multiuso do Instituto de Matemática e Estatística) - Bloco 1F (Campus Santa Mônica) da Universidade Federal de Uberlândia e por videoconferência, a Banca Examinadora, designada pelo Colegiado do Programa de Pós-graduação em Matemática, assim composta: Professores Doutores: Moiseis dos Santos Cecconello - Universidade Federal do Mato Grosso/UFMT, Josuel Kruppa Rongenski - Universidade Federal de Uberlândia / UFU e Rosana Sueli da Motta Jafelice - Instituto de Matemática e Estatística - IME/UFU orientadora da candidata.

Iniciando os trabalhos a presidente da mesa, Dra. Rosana Sueli da Motta Jafelice , apresentou a Comissão Examinadora e a candidata, agradeceu a presença do público, e concedeu à discente a palavra para a exposição do seu trabalho.

A duração da apresentação da discente e o tempo de arguição e resposta foram conforme as normas do Programa. A seguir a senhora presidente concedeu a palavra, pela ordem sucessivamente, aos examinadores, que passaram a arguir a candidata. Ultimada a arguição, que se desenvolveu dentro dos termos regimentais, a Banca, em sessão secreta, atribuiu o resultado final, considerando a candidata:

Aprovada.

Esta defesa faz parte dos requisitos necessários à obtenção do título de Mestre.

O competente diploma será expedido após cumprimento dos demais requisitos, conforme as normas do Programa, a legislação pertinente e a regulamentação

interna da UFU.

Nada mais havendo a tratar foram encerrados os trabalhos. Foi lavrada a presente ata que após lida e achada conforme foi assinada pela Banca Examinadora.

Documento assinado eletronicamente por **Rosana Sueli da Motta Jafelice**, **Professor(a) do Magistério Superior**, em 25/02/2026, às 16:05, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do Decreto nº 8.539, de 8 de outubro de 2015.

Documento assinado eletronicamente por **Josuel Kruppa Rogenski**, **Professor(a) do Magistério Superior**, em 25/02/2026, às 16:05, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do Decreto nº 8.539, de 8 de outubro de 2015.

Documento assinado eletronicamente por **Moiseis dos Santos Cecconello**, **Usuário Externo**, em 25/02/2026, às 16:05, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do Decreto nº 8.539, de 8 de outubro de 2015.

A autenticidade deste documento pode ser conferida no site https://www.sei.ufu.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **7071375** e o código CRC **D7920D5A**.

Referência: Processo nº 23117.009483/2026-43 SEI nº 7071375

Dedicatória

Dedico este trabalho às minhas ávos, Manoelita Barcelos Flor e Zélia dos Santos, meus maiores exemplos de força, coragem e amor. Sei que estarão sempre ao meu lado, me guiando me dando forças em cada momento. Obrigada por sempre cuidarem de mim e por me tornarem a mulher que sou hoje.

Agradecimentos

Dedico este espaço para agradecer a cada pessoa que esteve ao meu lado, me apoiando durante minha jornada acadêmica. Todas foram de imensa importância para a conclusão deste trabalho.

Em primeiro lugar, agradeço a Deus e aos meus pais, Alexsandra Lúcia de Andrade e Kleber Barcelos Flor, que sempre acreditaram em mim e me incentivaram em cada escolha. Obrigada por serem meu maior exemplo de força e dedicação.

À minha orientadora, Dra. Rosana Sueli da Motta Jafelice, agradeço por cada ensinamento, seja matemático ou de vida. Sua visão, criatividade e determinação em relação à pesquisa me inspiraram todos os dias na realização deste trabalho. Sou muito grata por toda a paciência, apoio e confiança depositada em mim.

Ao meu coorientador, Prof. Dr. José Waldemar da Silva, agradeço por todo o suporte, ensinamentos e por cada ideia compartilhada, que foram essenciais para o desenvolvimento e a conclusão do estudo. Agradeço também ao professor da Faculdade de Educação Física e Fisioterapia da Universidade Federal de Uberlândia, Dr. Thiago Ribeiro Teles dos Santos, pelas reuniões enriquecedoras e por todo o conhecimento compartilhado sobre o contexto da fisioterapia esportiva.

À minha segunda família, Lívia, Samara, Lutiéle, Laura, Dhara e Alícia, obrigada por serem o meu sinônimo de casa em Uberlândia. Agradeço em especial à Maria Clara, por ser meu maior suporte em cada passo da minha vida, obrigada por cada palavra, por cada motivação em momentos difíceis e por sempre estar ao meu lado.

Aos meus colegas de mestrado, agradeço por cada momento de aprendizado, ensinamento e apoio mútuo. Agradeço em especial ao Mateus Fernando, Victor Patrick, Silas, Ially, Matheus Felipe, Carlos, Marcos, José Armando, Mateus Deodato e Marlennhi.

Agradeço também a todas as que foram minhas colegas de time, pelos ensinamentos dentro e fora de quadra. Obrigada por cada momento, cada treino e campeonato vivido juntas. Vocês me ensinaram o verdadeiro significado de parceria e amizade. Obrigada por tornarem minha jornada acadêmica mais leve e divertida.

Expresso meu profundo agradecimento às instituições que apoiaram este trabalho. Ao Laboratório de Cálculo Numérico e Simbólico (LCNS), por todo o suporte técnico essencial para o processamento dos algoritmos utilizados neste trabalho. E à FAPEMIG, pela bolsa de incentivo à pesquisa, que possibilitou a realização desta investigação e dos meus estudos no Mestrado da Universidade Federal de Uberlândia.

Por fim, agradeço a todas as pessoas que, de alguma forma, contribuíram para a minha chegada até este momento. Cada palavra de incentivo, cada gesto de apoio e cada momento compartilhado foram fundamentais para a conclusão desta etapa da minha vida.

Resumo

O treinamento físico de atletas alavanca o desempenho esportivo em busca da alta performance, porém esse contexto pode desencadear efeitos adversos que prejudicam estes profissionais e as organizações esportivas. As lesões desportivas emergem de uma rede complexa de fatores, entretanto sua ocorrência está fortemente associada a volumes inadequados de cargas de treinamento. O objetivo deste trabalho é realizar a predição da indicação e do risco de lesões esportivas em atletas de futebol feminino profissional, sendo a indicação de lesão um valor binário (0 ou 1) e o risco um valor entre 0 e 1. Neste estudo foram utilizados 11.612 registros diários de atletas de dois clubes profissionais de futebol feminino da Noruega, coletados através do aplicativo Player Monitoring System (pmSys) durante as temporadas de 2020 e 2021. O banco de dados utilizado conta com variáveis relacionadas à carga de treinamento, bem-estar e ocorrência de lesões. As variáveis de carga de treinamento foram coletadas através da escala de Percepção Subjetiva de Esforço (PSE), uma ferramenta que quantifica a intensidade de um exercício com base na percepção da atleta. Nos dados, as lesões são registradas no dia ocorrido, sendo considerada a variável de saída do modelo. Os modelos de predição de indicação de lesão foram desenvolvidos utilizando os métodos de aprendizado de máquina Árvore de Decisão e Floresta Aleatória. As métricas utilizadas para avaliar o desempenho dos modelos foram sensibilidade (taxa de acerto das predições positivas) e acurácia (taxa de acerto de todas as predições). A Floresta Aleatória superou a Árvore de Decisão, apresentando sensibilidade de 64,52% e acurácia de 98,71%. Para a predição do risco de lesão, foram aplicados os métodos neuro-fuzzy: Sistema de Inferência Neuro-Fuzzy Dinâmico Evolutivo, do inglês *Dynamic Evolving Neural Fuzzy Inference System* (DENFIS) e o Método de Agrupamento Subtrativo, do inglês *Subtractive Clustering Method* com Fuzzy C-Means (SBC/FCM). Cada modelo utiliza quatro variáveis de entrada, relacionadas ao treinamento e bem-estar da atleta, para gerar um Sistema Baseado em Regras Fuzzy (SBRF) com o intuito de predizer o risco de lesão da atleta. No conjunto de teste, o modelo DENFIS apresentou sensibilidade de 75% e acurácia de 85,38%, enquanto o modelo SBC/FCM obteve sensibilidade de 75% e acurácia de 91,70%, na validação cruzada do tipo k-*fold* com $k = 5$, o modelo DENFIS alcançou maiores porcentagens nos *folds* na métrica sensibilidade. Nos modelos de risco de lesão, para tratar o desbalanceamento do atributo "lesão", uma vez que sua ocorrência, em geral, é de baixa frequência, foi aplicada a técnica SMOTE, do inglês *Synthetic Minority Over-sampling Technique*, que gera dados sintéticos da classe minoritária (lesão). Neste estudo também foi desenvolvido uma interface computacional através do *software* R que calcula o risco de lesão do usuário, a partir de dados de treinamento físico e bem-estar, utilizando o modelo obtido pelo DENFIS.

Palavras-chave: Inteligência Artificial; Modelos Preditivos; Risco de Lesão; Sistemas Baseados em Regras Fuzzy; Clusterização.

Abstract

Physical training plays a key role in enhancing athletes' performance in the pursuit of high-level results; however, this context can trigger adverse effects that harm both these professionals and sports organizations. Sports injuries emerge from a complex network of factors, however, their occurrence is strongly associated with inadequate training load volumes. The aim of this work is to predict the indication and risk of injuries in professional female soccer athletes, where the injury indication is a binary value (0 or 1) and the risk is a value between 0 and 1. In this study, 11,612 daily records from athletes of two professional female soccer clubs in Norway were used, collected via the Player Monitoring System (pmSys) application during the 2020 and 2021 seasons. The database used includes variables related to training load, well-being, and injury occurrence. Training load variables were collected using the Rating of Perceived Exertion (RPE) scale, a tool that quantifies exercise intensity based on the athlete's perception. In the dataset, injuries were recorded on the day they occurred, serving as the model's output variable. Injury indication prediction models were developed using Decision Tree and Random Forest machine learning methods. The metrics used to evaluate model performance were sensitivity (success rate of positive predictions) and accuracy (success rate of all predictions). Random Forest outperformed Decision Tree, presenting a sensitivity rate of 64.52% and an accuracy rate of 98.71%. To predict injury risk, neuro-fuzzy methods were applied: the Dynamic Evolving Neural Fuzzy Inference System (DENFIS) and the Subtractive Clustering Method with Fuzzy C-Means (SBC/FCM). Each model employs four training and well-being–related input variables to generate a Fuzzy Rule-Based System (FRBS) aimed at predicting the athlete's injury risk. In the test set, the DENFIS model presented a sensitivity rate of 75% and an accuracy rate of 85.38%, while the SBC/FCM model obtained a sensitivity of 75% and an accuracy of 91.70%. Using k-fold cross-validation with $k = 5$, the DENFIS model achieved higher percentages across the folds in the sensitivity metric. In the injury risk models, to address the imbalance of the "injury" attribute, since its occurrence is generally of low frequency, the Synthetic Minority Over-sampling Technique (SMOTE) was applied to generate synthetic data for the minority class (injury). This study also developed a computational interface using R software that calculates the user's injury risk based on physical training and well-being data, utilizing the model obtained by DENFIS.

Keywords: Artificial Intelligence; Predictive Models; Injury Risk; Fuzzy Rule-Based Systems; Clustering.

Lista de Figuras

Lista de Tabelas

Sumário

Introdução

"Em Deus nós confiamos; todos os outros devem trazer dados." (W. Edwards Deming)

Atualmente, o futebol é o esporte mais popular do mundo (CUSTÓDIO, 2011), sendo praticado por mais de 265 milhões de pessoas de todas as idades e gêneros, segundo uma pesquisa realizada pela Federação Internacional de Futebol (FIFA) em 2006 (CONMEBOL, 2013). O futebol moderno surgiu na Inglaterra em 1863, porém, foi apenas durante a Primeira Guerra Mundial (1914-1918) que o futebol feminino começou a se popularizar no país, uma vez que, com os homens nos campos de batalha, as partidas de futebol entre mulheres eram necessárias para arrecadar dinheiro durante a guerra (PIZARRO, 2024). Entretanto, em 1921, a Federação Inglesa de Futebol, do inglês *The Football Association* (FA) proibiu o futebol feminino. No Brasil, o futebol feminino também passou por um período de proibição em 1941, que durou cerca de 40 anos, tendo sua liberação por lei apenas na década de 80 (PIZARRO, 2024), (ZULQUES, 2023).

Em 2023, no último relatório oficial da FIFA sobre o futebol feminino, *FIFA Women's Football: Member Associations Survey Report 2023*, foi apontado que o futebol feminino é praticado por mais de 16 milhões de mulheres em todo o mundo (FIFA, 2023). Representando um aumento de aproximadamente 24% em relação ao número de jogadoras registrado em 2019. A modalidade vem crescendo e se profissionalizando significativamente nos últimos anos. A próxima Copa do Mundo Feminina da FIFA, que será realizada no Brasil em 2027, contará com a participação de 32 seleções nacionais, um aumento em relação às 16 seleções que participaram do torneio de 2011, realizado na Alemanha (Museu do Futebol, 2011).

Com o aumento da popularidade do futebol feminino, o número de clubes também cresceu consideravelmente. Atualmente, há 55.622 clubes de futebol feminino em todo o mundo (FIFA, 2023). A partir da profissionalização da modalidade, surge a necessidade de aprimorar a performance esportiva das atletas através de treinamentos mais intensos e específicos, visando alcançar melhores resultados em competições.

O treinamento físico de atletas é de fundamental importância para seu desempenho e faz parte de sua rotina diária. No entanto, o número de lesões associado à prática esportiva é expressivo (EMERY; PASANEN, 2019), impactando não apenas a saúde e o desempenho das atletas, mas também gerando custos financeiros às organizações esportivas, visto que muitos atletas lesionados são tratados no departamento médico do próprio clube. Além disso, ao mesmo tempo em que a ausência de atletas devido à lesões pode comprometer a equipe em campo, é importante pontuar que a exposição excessiva a jogos e treinamentos pode agravar lesões pré-existentes, prejudicando a vida e a carreira das atletas. Compreende-se que lesões esportivas são o resultado de uma combinação complexa de fatores (EETVELDE *et al.*, 2021). Entretanto, as lesões, em geral, podem ocorrer quando os atletas estão expostos a cargas inadequadas de treinamento e competição (WINDT; GABBETT, 2017). Pesquisas recentes sugerem que as

cargas de treinamento do atleta são um importante fator de risco na investigação de lesões (SOLIGARD *et al.*, 2016). Surge, portanto, a necessidade de monitorar as atletas para tentar identificar previamente possíveis lesões, contribuindo para sua performance esportiva e sua saúde.

A escala de Percepção Subjetiva de Esforço (PSE) da sessão é uma forma válida de quantificar o treinamento em diversos tipos de esportes e exercícios (FOSTER *et al.*, 2001). O método é uma ferramenta subjetiva utilizada para quantificar a intensidade de um exercício, considerando a percepção do atleta em uma escala de 1 a 10. Por exemplo, uma atleta em seu treinamento do dia pode relatar que a intensidade de seu treino de 60 minutos foi de 7 na escala de PSE, indicando uma alta intensidade. O valor do PSE dessa sessão (PSEs) é calculado multiplicando a intensidade 7 pela duração do exercício, resultando em 420 unidades de carga de treinamento (7 x 60 minutos = 420). Em um modelo de previsão de lesões realizado por Gabbett (GABBETT, 2010), utilizando o PSEs, foi mostrado que jogadores que excederam o limite de carga de treinamento semanal planejada para uma determinada etapa da competição tiveram 70 vezes mais probabilidade de apresentar lesões. A ferramenta PSE também se destaca pela praticidade em sua aplicação e seu baixo custo, uma vez que sua implementação requer apenas formulários aplicados diariamente às atletas, facilitando a coleta de dados por parte de fisioterapeutas e preparadores físicos.

A partir do PSE, é possível calcular outras medidas de carga de treinamento, como a Carga Aguda de Treinamento (CAT) e a Carga Crônica de Treinamento (CCT). Em um estudo realizado por (WINDT; GABBETT, 2017), foi demonstrado que a relação entre essas duas cargas está fortemente associada ao risco de lesão em atletas. Além disso, outras medidas importantes derivadas do PSE são a monotonia e a tensão do treinamento, as quais indicam que a forma como a carga de treinamento do atleta é distribuída ao longo da semana é tão importante quanto a sua carga total de treinamento. Essas variáveis possuem um forte potencial para indicar desfechos adversos do treinamento, como lesões (FOSTER *et al.*, 2001). Um treinamento intenso com pouca variação de carga na semana (baixo desvio padrão) indica alta monotonia. Uma pesquisa realizada com atletas universitárias de futebol feminino mostrou que a alta monotonia está associada a um aumento na incidência de lesões e baixo desempenho esportivo (PUTLUR *et al.*, 2004). Em um estudo anterior, Foster (1998) observou que 77% das lesões estavam associadas a um pico prévio na monotonia, enquanto 89% puderam ser explicadas por um aumento precedente na tensão do treinamento. Além da carga de treinamento, a duração do sono dos atletas é uma variável que precisa ser monitorada. De acordo com (HARALDSDOTTIR *et al.*, 2021), uma maior duração do sono está associada a um risco menor de lesões.

A abordagem do aprendizado de máquina, subcampo da inteligência artificial, tem sido proposta como uma ferramenta na ciência do esporte para investigar interações complexas entre diferentes preditores e identificar padrões em variáveis, com o propósito final de predição (BITTENCOURT *et al.*, 2016). No ano de 2022, Jauhiainen e Kauppi realizaram um estudo sobre a predição de lesões do ligamento cruzado anterior em atletas de handebol e futebol feminino de elite, utilizando métodos de aprendizado de máquina (JAUHIAINEN *et al.*, 2022). Nesse estudo, foram utilizadas variáveis biomecânicas coletadas por meio de testes como o salto vertical, a prova de equilíbrio unipodal e testes de força musculoesquelética, além disso, foi utilizada a técnica SMOTE para gerar dados sintéticos da classe minoritária visando o balanceamento no conjunto de dados de treinamento.

Outra subárea da inteligência artificial que tem ganhado destaque na modelagem preditiva são as redes neuro-fuzzy, arquiteturas híbridas que combinam redes neurais artificiais com SBRF. A teoria dos conjuntos fuzzy, introduzida por Lotfi A. Zadeh em 1965, surge para lidar com a imprecisão e a incerteza em modelos matemáticos (JAFELICE; BARROS; BASSANEZI, 2023). A lógica fuzzy considera termos linguísticos como “aproximadamente”, “em torno de”, dentre outros, aproximando modelos da realidade. Dessa forma, as redes neuro-fuzzy combinam

a capacidade de aprendizado e reconhecimento de padrões das redes neurais com SBRF, que incorporam o conhecimento humano e realizam a inferência e a tomada de decisão (JANG; SUN; MIZUTANI, 1997), sendo amplamente utilizadas em diversas áreas, como engenharia, economia, biologia, medicina, entre outras.

No cenário da modelagem preditiva de lesões esportivas com redes neuro-fuzzy, Patel *et al.* (2025) apresentaram um modelo preditivo para lesões de epicondilite lateral (cotovelo de tenista) utilizando Redes Neurais Artificiais (RNA) e Sistema de Inferência Neuro-Fuzzy Adaptativos (ANFIS) (PATEL *et al.*, 2025). O modelo utiliza variáveis biomecânicas como ângulo de flexão, torque e força do cotovelo para prever o rompimento em milímetros do tendão Extensor Carpi Radialis Brevis, sendo a saída uma variável contínua. O ANFIS é considerado um dos sistemas pioneiros na área de redes neuro-fuzzy, sendo utilizado em diversas aplicações (JANG, 1993).

No contexto de pesquisas sobre predição de lesões esportivas utilizando variáveis relacionadas à PSE, destacam-se o trabalho de Rogalski, Dawson, Heasman e Gabbett, publicado em 2013. O estudo tem como objetivo o desenvolvimento de um modelo de predição de lesões em atletas de futebol australiano, utilizando variáveis de carga de treinamento coletadas por meio da escala de percepção de esforço PSE (ROGALSKI *et al.*, 2013). A modelagem estatística (regressão logística) foi utilizada para relacionar cargas e mudanças de carga com a incidência de lesões. Também é apresentado o trabalho de Carey e colaboradores, também sobre modelos de predição de lesão em atletas de futebol australiano utilizando variáveis coletadas pelo método PSE e por dispositivos *Global Positioning System* (GPS) (CAREY *et al.*, 2018). O estudo foi realizado em 2018 e os modelos foram desenvolvidos por meio de métodos de aprendizado de máquina, como regressão logística, florestas aleatórias (BREIMAN, 2001) e máquinas de vetor de suporte.

Nessa perspectiva, ressalta-se que a modelagem preditiva não deve se limitar apenas a prever a ocorrência de uma lesão em si, mas, se possível, buscar identificar o risco de lesão em um nível individual (MEEUWISSE *et al.*, 2007). Dessa forma, o objetivo geral desta pesquisa inclui:

- Desenvolver modelos de predição de indicação de lesão esportiva (saída binária 0 ou 1) utilizando os métodos de aprendizado de máquina Árvore de Decisão e Floresta Aleatória;

- Desenvolver modelos de predição do risco de lesão esportiva (saída entre 0 e 1) utilizando o Sistema de Inferência Neuro-Fuzzy Dinâmico Evolutivo, do inglês *Dynamic Evolving Neural Fuzzy Inference System* (DENFIS), e o Método de Agrupamento Subtrativo, do inglês *Subtractive Clustering Method* com Fuzzy C-Means (SBC/FCM). Cada um desses métodos gera um Sistema Baseado em Regras Fuzzy (SBRF) para a predição do risco de lesão.

Nesse sentido, para alcançar o objetivo geral, foram estabelecidos os seguintes objetivos específicos:

- Estudar e exemplificar métodos de aprendizado de máquina, além de implementá-los computacionalmente atráves do *software* R (R Core Team, 2024), por meio do ambiente de desenvolvimento integrado RStudio (Posit team, 2025);

- Compreender a utilização do índice de Gini no aprendizado do método árvore de decisão;

- Utilizar a técnica SMOTE na variável de saída (ocorrência de lesão) para balancear o conjunto de dados utilizado, uma vez que "lesões" é uma variável com alto desbalanceamento;

- Realizar uma análise dos resultados obtidos pela Árvore de Decisão e a Floresta Aleatória;

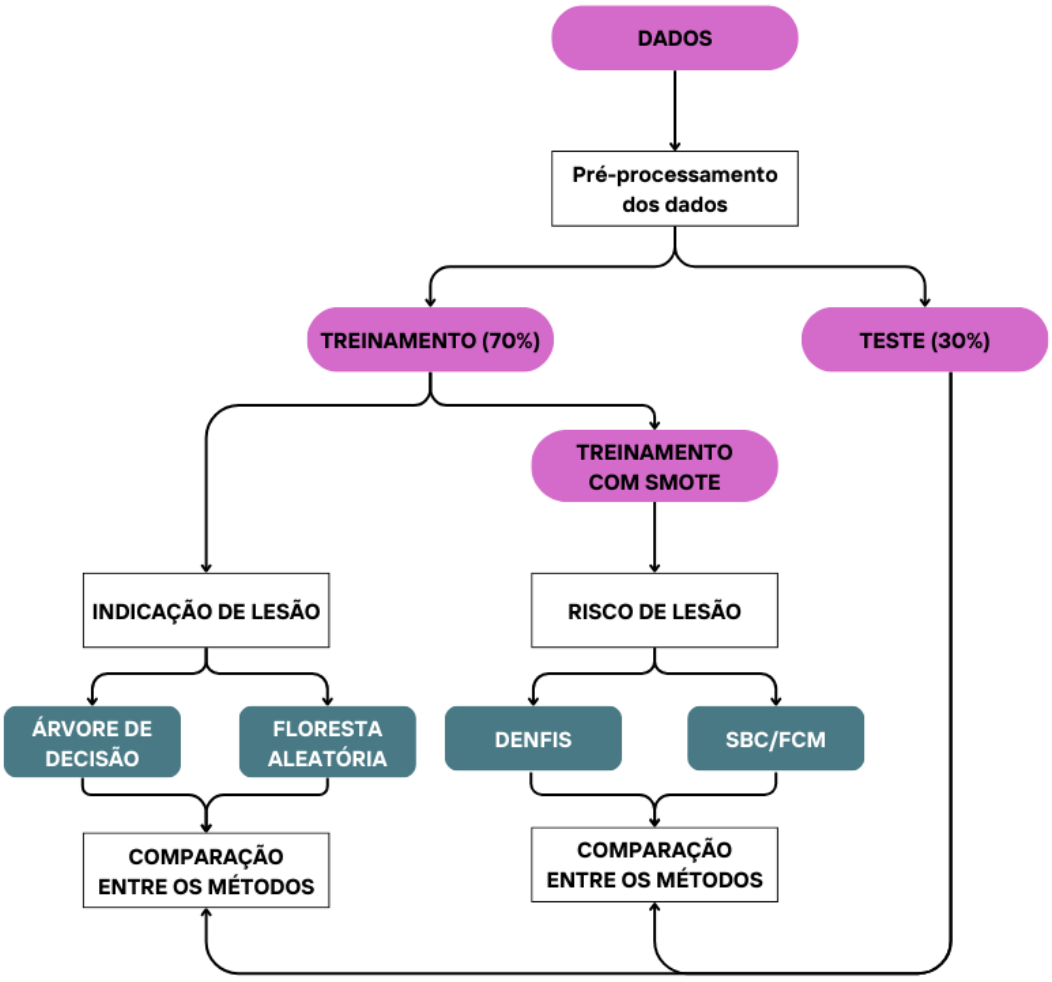
- Selecionar a melhor combinação de variáveis de entrada para os modelos desenvolvidos, buscando apontar quais das variáveis disponíveis possuem maior potencial na predição de lesões esportivas em atletas de futebol feminino;

- Realizar uma comparação entre os sistemas de inferência fuzzy DENFIS e SBC/FCM atráves dos resultados obtidos;

- Produzir uma interface computacional que calcula o risco de lesão do usuário, a partir de dados de treinamento físico de bem-estar.

A metodologia adotada tanto nos modelos de indicação de lesão quanto nos modelos de predição de risco foi a mesma. Na Figura 1 está esquematizada a metodologia deste trabalho.

Figura 1: Esquema da metodologia da pesquisa.



Fonte: Autoria própria.

Na Figura 1, os blocos rosas correspondem aos conjuntos de dados, os blocos verdes indicam os métodos utilizados, enquanto os retângulos pretos atuam como blocos auxiliares para a interpretação do esquema. O banco de dados utilizado neste trabalho foi coletado atráves de uma pesquisa de monitoramento com atletas de dois times de futebol feminino profissional da Noruega (MIDOGLU *et al.*, 2024). Os dados foram submetidos a uma etapa de pré-processamento, visando a exclusão de registros duplicados e o tratamento de outras inconsistências detalhadas

no Capítulo 3. A próxima etapa consistiu na separação dos dados em treino e teste, designando 70% para treinamento e 30% para teste. O fluxo divide-se em dois blocos: "indicação de lesão" e "risco de lesão". O primeiro bloco compreende os modelos de indicação de lesão, cuja saída é uma variável binária ("lesão" ou "não lesão"). Nessa etapa, utilizam-se os métodos de classificação Árvore de Decisão e Floresta Aleatória, sendo realizada a comparação entre eles a partir do conjunto de teste. O segundo bloco consiste nos modelos de risco de lesão, cuja saída é contínua (valor entre 0 e 1). Nesta fase foi aplicada a técnica SMOTE no conjunto de treinamento para gerar dados sintéticos da classe minoritária, visando o balanceamento entre as classes "lesão" e "não lesão". Neste bloco, foram empregados os métodos DENFIS e SBC/FCM, e realizada uma análise comparativa entre ambos.

Este estudo pode ter contribuições relevantes para o campo da predição de lesões esportivas, se diferenciando de outros trabalhos da área por utilizar dados de futebol feminino profissional, fazendo uso de variáveis relacionadas à PSE, sendo de fácil coleta e baixo custo, além de empregar métodos de aprendizado de máquina e sistemas neuro-fuzzy via clusterização na modelagem preditiva. Os modelos apresentados podem ser aplicados em diversos contextos esportivos, em qualquer time de futebol feminino com as mesmas características e condições, auxiliando preparadores físicos e fisioterapeutas na tomada de decisões relacionadas ao treinamento e à prevenção de lesões de atletas. Além disso, a metodologia realizada neste trabalho pode servir como base para futuras pesquisas na área, incentivando o desenvolvimento de modelos de predição mais avançados para diferentes modalidades esportivas.

A escrita deste trabalho está estruturada em três capítulos, além desta introdução e das considerações finais. No Capítulo 1, são apresentados conceitos iniciais sobre aprendizado de máquina, como *overfitting*, validação cruzada e tratamento de dados desbalanceados. Além disso, o principal foco deste capítulo é a apresentação dos métodos de aprendizado de máquina utilizados neste trabalho: Árvore de Decisão, Floresta Aleatória e Redes Neurais.

O Capítulo 2 aborda os principais conceitos da teoria dos conjuntos fuzzy e introduz os Sistemas Baseados em Regras Fuzzy (SBRF). São apresentados também os Sistemas de Inferência Neuro-Fuzzy via Clusterização DENFIS e o SBC/FCM. Ao final do capítulo, exemplifica-se o processo de criação dos clusters a partir de um conjunto de dados, além da geração da parte antecedente das regras fuzzy e suas funções de pertinência. O exemplo didático foi realizado em ambos os métodos, DENFIS e SBC/FCM, para facilitar a compreensão do leitor sobre o funcionamento dos mesmos.

No Capítulo 3, é detalhada a metodologia adotada na pesquisa, incluindo a descrição do conjunto de dados utilizado, o pré-processamento dos dados e a implementação dos modelos. A apresentação dos modelos e seus resultados é dividida em duas seções: modelos de aprendizado de máquina para predição da indicação de lesão e modelos de sistemas neuro-fuzzy para predição do risco de lesão. Cada seção apresenta os detalhes dos modelos implementados, os hiperparâmetros e variáveis utilizadas, a avaliação do desempenho dos modelos utilizando as métricas sensibilidade e acurácia, além da análise dos resultados obtidos. Ao final de cada seção, é realizada uma comparação entre cada método.

Para encerrar, o Capítulo 4 apresenta as considerações finais sobre o estudo realizado, destacando as principais conclusões, vantagens e limitações dos modelos desenvolvidos durante a pesquisa, além de sugestões para trabalhos futuros.

Capítulo 1

Aprendizado de Máquina

"A IA é a arte de fazer com que computadores façam coisas inteligentes." (Waldrop, 1987)

1.1 Conceitos Iniciais

Em 1959, Arthur Samuel, conceituado como um dos pioneiros do campo de aprendizagem de máquina, a definiu como sendo "a habilidade de aprender sem ser explicitamente programada" (SAMUEL, 1959). O Aprendizado de Máquina é considerado um subcampo da área de Inteligência Artificial (IA) dedicado ao estudo de técnicas computacionais de aprendizado. Esse ramo desenvolve sistemas que possuem a capacidade de aprender a partir de experiências, utilizando dados provenientes de soluções do problema em questão.

Por exemplo, o filtro de *spam* da caixa de *e-mails* é um algoritmo criado utilizando técnicas de aprendizado de máquina. O filtro analisa diversos padrões de *e-mails* considerados *spam* para identificar futuras mensagens como sendo ou não *spam*. Nesse processo, são considerados diversos fatores, como palavras-chave, assunto, remetente, corpo do *e-mail* e as ações do usuário ao marcar algum *e-mail* como *spam*. Recomendações de conteúdo em plataformas como *YouTube* e *Netflix*, assistentes virtuais como *Siri* e *Alexa*, tradução automática e reconhecimento facial são outros exemplos em que o aprendizado de máquina se faz presente no contexto atual.

Os sistemas de aprendizado de máquina são separados por diversas características; uma delas é o tipo e a quantidade de supervisão que o sistema recebe durante a sua fase de treinamento (REZENDE, 2003). Para facilitar a compreensão, é destacada a diferença entre os conceitos de algoritmo, método e modelo. Por exemplo, dado um conjunto de dados, pode-se adotar o método de árvore de decisão para a construção do modelo. Nesse contexto, o algoritmo implementado processa os dados e gera uma árvore de decisão final. Essa árvore, configurada com os parâmetros aprendidos a partir dos dados, constitui o modelo resultante.

- Aprendizado de máquina supervisionado

  Os métodos supervisionados são métodos treinados com dados rotulados, ou seja, que possuem a solução do sistema (rótulos) como uma das entradas. O objetivo do algoritmo supervisionado é construir um classificador para determinar a classe de novos dados que ainda não possui um rótulo (REZENDE, 2003). Para rótulos de variável contínua o problema é definido como regressão, enquanto para valores discretos, é considerado um problema de classificação. Um exemplo de modelo supervisionado são sistemas de segurança baseados em detecção de imagens, para identificar por exemplo, se uma imagem possui ou não um carro. Nesse sistema, entre os dados de treinamento considere imagens

de carros e não carros, é necessário informar ao algoritmo se a imagem possui ou não o objeto, para que este possa identificar padrões e classificar novas imagens a partir destes. Alguns exemplos de método supervisionados são regressão linear, regressão logística, árvore de decisão, florestas aleatórias e redes neurais.

- Aprendizado de máquina não supervisionado

  Os métodos não supervisionados são treinados com dados não rotulados, onde não conhecemos a saída ou solução do modelo. Nessa categoria, o programa analisa exemplos fornecidos pelo banco de dados e busca identificar se alguns deles podem ser agrupados por um padrão, formando agrupamento ou *clusters* (GÉRON, 2017). Após essa etapa, os clusters passam por uma análise para determinar seu significado de acordo com o problema em questão (REZENDE, 2003). Por exemplo, em algoritmos de visualização em redes sociais é necessário separar usuários por grupos de acordo com suas características como gostos, tempo de tela, histórico de visualizações entre outros. O cientista não possui a informação de qual grupo cada usuário está (rótulo), uma vez que o rótulo é definido a partir das atitudes tomadas pelo próprio usuário ao utilizar a rede social. O algoritmo encontra tais conexões e similaridades entre os usuários sem o auxilio do programador e assim os rotula, com o intuito de entregar conteúdos de acordo com cada grupo.

- Aprendizado semissupervisionado

  Nos métodos semissupervisionados existe a presença de dados rotulados e dados não rotulados durante o treinamento, sendo, no geral, a maior parte dos dados não rotulados. Esse tipo de aprendizado pode ser uma solução para quando não se tem uma quantidade de dados rotulados o suficiente para utilizar um algoritmo de aprendizado supervisionado ou quando há um alto custo em rotular os dados. O modelo utilizado pelo *Google Photos* é um exemplo de aprendizado semissupervisionado. Ao inserir um conjunto de fotos no aplicativo, o mesmo reconhece que a pessoa A apareceu em um certo conjunto de fotos, mas sem nomear esta pessoa. Essa é a parte não supervisionada do algoritmo, chamada de *clustering* (agrupamento) (GÉRON, 2017). A parte supervisionada é realizada pelo próprio usuário, onde o mesmo reconhece e nomeia a pessoa em apenas uma foto, e com isso o algoritmo irá identificar e nomear a pessoa em todas as imagens restantes.

- Aprendizado por reforço

  O aprendizado por reforço se assemelham ao aprendizado supervisionado, porém sem um conjunto de dados de treinamento. Nesses métodos o algoritmo observa o ambiente e toma suas próprias decisões, e a partir delas recebe "recompensas" ou "penalidades" para decidir se tal decisão foi ou não bem tomada (GÉRON, 2017). Assim, o algoritmo constrói uma solução mais otimizada, utilizando de tentativas e erros para assim, aprender por si só a melhor estratégia. Exemplos de aprendizado por reforço são algoritmos presentes em robôs para aprenderem a andar ou em computadores para ganhar um determinado jogo.

Há também a diferenciação entre métodos por on-line e off-line.

Métodos On-line x Métodos Off-line

Métodos off-line, também conhecidos como métodos em lote, são aqueles que processam o conjunto de treinamento inteiro de uma única vez, como a árvore de decisão, que precisa examinar todas as amostras para encontrar uma divisão ótima para cada nó da árvore. Redes neurais tradicionais também são exemplos de métodos off-line. Tais métodos, em problemas com uma grande quantidade de dados, podem ter um custo computacional muito alto, sendo inviáveis para desafios de dados em larga escala (BISHOP, 2006).

Em contrapartida aos métodos off-line, existem os métodos on-line, também conhecidos como métodos sequenciais. Em técnicas on-line, o modelo é treinado de forma sequencial, adaptando sua estrutura e parâmetros a cada dado que entra no modelo. Podem ser usadas no aprendizado em tempo real, onde um fluxo constante de dados é fornecido como entrada do algoritmo e previsões precisam ser realizadas antes que todos os dados sejam visualizados (BISHOP, 2006). Um exemplo do uso dessa técnica são algoritmos de recomendações de conteúdo. A cada iteração do usuário em uma plataforma, é gerada uma pequena quantidade de dados. O diferencial do aprendizado on-line acontece nesta etapa; o algoritmo sequencial não espera por um "lote", uma grande quantidade de dados, como em algoritmos off-line. Ele utiliza cada nova iteração do usuário para ajustar, naquele mesmo instante, os parâmetros do modelo e, assim, gerar a saída: a recomendação instantânea de um conteúdo para o usuário.

1.1.1 Treino e Teste do Modelo

Na linguagem do aprendizado de máquina, o desenvolvimento de um modelo supervisionado é dividido em duas etapas chamadas de treinamento e teste. Inicialmente, todo o conjunto de dados é dividido em um conjunto de treinamento e um conjunto de teste. A proporção dessa divisão é definida pelo programador e pode variar de acordo com o tamanho do conjunto de dados e a natureza do problema (BISHOP, 2006). Tanto as observações destinadas ao conjunto de treinamento quanto as destinadas ao conjunto de teste são selecionadas, em geral, de maneira aleatória pelo *software*.

A etapa de treinamento é o momento de estimar os parâmetros do modelo. Nessa fase, o algoritmo interpreta os dados do conjunto de treinamento na tentativa de identificar padrões e relações nos dados para, assim, poder realizar uma previsão. Após desenvolvido, inicia-se a etapa de testagem, onde os dados do conjunto de teste, que possuem seu rótulo original e não foram utilizados para treinar o modelo, são aplicados ao modelo, obtendo um novo rótulo.

Assim, é possível comparar os novos rótulos preditos pelo modelo com o rótulo original de cada observação do conjunto de teste, avaliando assim o desempenho deste modelo. Para isso, são utilizadas funções matemáticas que auxiliam na tarefa de avaliar a capacidade de erro e acerto de um modelo. Tais funções são denominadas métricas.

1.1.2 Algumas Métricas

As métricas apresentadas a seguir são específicas para problemas de classificação, uma vez que os modelos a serem desenvolvidos neste trabalho são dessa natureza. As definições foram obtidas das referências (GÉRON, 2017) e (WITTEN; FRANK; HALL, 2011).

Definição 1.1. *A medida Acurácia ou Taxa de Acerto é definida pela razão entre o número de previsões corretas e o número total de previsões.*

$$Acurácia = \frac{Número\ de\ previsões\ corretas}{Total\ de\ previsões}. \tag{1.1}$$

Algumas métricas dão enfoque no custo de se fazer uma predição errada para uma determinada classe, a depender do problema, o custo de um falso positivo é maior do que o de um falso negativo, ou ao contrário.

Considere o exemplo de um modelo de imagem que indica a existência ou não de uma mancha de óleo no mar. O custo para o meio ambiente, de um falso alarme da presença de óleo em determinado local do oceano, é muito menor do que classificar a mancha como "não óleo", ou seja, da mancha passar despercebida na natureza (WITTEN; FRANK; HALL, 2011).

No tema de aplicação deste trabalho ocorre o mesmo. Quando se trata da indicação de lesões esportivas em atletas, a previsão atua como um auxiliar no processo de prevenção, funcionando

como um alerta para que fisioterapeutas e a equipe médica direcionem maior atenção à saúde de determinado atleta. Nesse contexto, o custo de identificar errado um atleta que não há chances de lesionar é menor do que o custo de ignorar um atleta que está prestes a ter uma lesão.

Em problemas em que há duas classes como saída, da mesma forma que indicação ou não de lesão, em uma predição há 4 possibilidades diferentes: os verdadeiros positivos (VP), verdadeiros negativos (VN), falsos positivos (FP) e falsos negativos (FN). Após o modelo realizar a previsão do conjunto de teste, é possível visualizar a quantidade de cada VP, VN, FP e FN em uma tabela, a qual é denominada matriz de confusão. Na Tabela 1.1 é apresentada a estrutura da Matriz de Confusão.

Tabela 1.1: Matriz de Confusão.

Classe Real	**Classe Predita**	
	Falso	**Verdadeiro**
Falso	Verdadeiros Negativos (VN)	Falsos Positivos (FP)
Verdadeiro	Falsos Negativos (FN)	Verdadeiros Positivos (VP)

Os erros de uma predição são a soma de falsos positivos e negativos, enquanto os acertos são a soma dos elementos da diagonal principal da matriz de confusão, os verdadeiros positivos e negativos. A depender da quantidade de dados, os resultados da matriz de confusão podem parecer desconexos (GÉRON, 2017), dessa forma, existem métricas que auxiliam na visualização do desempenho.

Definição 1.2. *A medida Precisão é definida como a "acurácia" das predições positivas, dada por:*

$$Precisão = \frac{VP}{VP + FP}. \tag{1.2}$$

Em um cenário onde o modelo acerte todas as previsões positivas, não prevendo nenhum falso positivo, é obtida uma precisão de 100% ($VP/(VP + 0) = 1$), entretanto o modelo pode ter errado nas previsões falsas. Dessa forma, a precisão geralmente é utilizada juntamente com outra métrica, chamada sensibilidade ou taxa positiva verdadeira.

Definição 1.3. *A medida Sensibilidade é definida como a proporção de observações positivas previstas de forma correta pelo modelo, dada por:*

$$Sensibilidade = \frac{VP}{VP + FN}. \tag{1.3}$$

Assim, quanto maior a sensibilidade, melhor é o desempenho do modelo na previsão de verdadeiros positivos. De fato, quando não há nenhum falso negativo, temos uma sensibilidade de 100% ($VP/(VP + 0) = 1$).

Portanto, a depender do modelo, é de extrema importância avaliar a precisão juntamente com a sensibilidade. Uma maneira simples de avaliar ambas as métricas uma única vez, é utilizar a medida F1-*Score*.

Definição 1.4. *A medida F1-Score é definida pela média harmônica entre a precisão e a sensibilidade, dada por:*

$$F1 = \frac{2}{\frac{1}{Precisao} + \frac{1}{Sensibilidade}}. \tag{1.4}$$

A métrica F1 pertence ao intervalo [0,1], sendo 1 o desempenho máximo do modelo. De fato, quando a precisão e a sensibilidade são iguais a 1 (desempenho perfeito), tem-se

$$F1 = \frac{2}{\frac{1}{1} + \frac{1}{1}} = 1. \tag{1.5}$$

Conforme mencionado anteriormente, em modelos de predição de lesão, um falso negativo é um erro com um peso maior que um falso positivo. Por esse motivo, neste trabalho é dado um alto enfoque à métrica sensibilidade para a avaliação dos modelos desenvolvidos.

A depender da divisão em treino e teste, além de outros fatores, o modelo pode acabar por sobreajustar os dados de treinamento, não conseguindo realizar boas previsões para novos dados. A seguir, é apresentado o conceito de *overfitting* e a técnica de validação cruzada, a qual auxilia na escolha do melhor modelo e pode evitar esse fenômeno.

1.1.3 Overfitting e Validação Cruzada

Suponha que você está viajando para outro país e um taxista te trate mal. Com isso, você pode acabar concluindo que todos os taxistas daquele país são grosseiros. Na teoria de aprendizado de máquina isso é chamado *overfitting* (GÉRON, 2017). Este fenômeno ocorre quando há um superajuste nos dados de treinamento, fornecendo uma boa previsão para esse conjunto, enquanto para novos dados este não fornece bons resultados. O *overfitting* ocorre quando o problema a ser solucionado é complexo demais para a quantidade de dados utilizada no algoritmo.

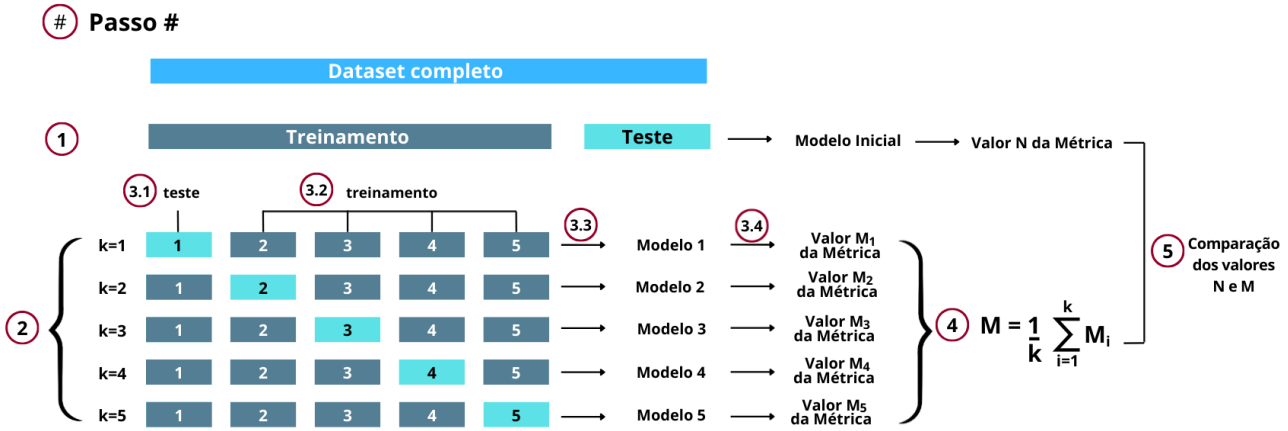
Uma possível solução para evitar o *overfitting* é utilizar outros hiperparâmetros no modelo. Hiperparâmetros são parâmetros definidos antes do treinamento do modelo, como por exemplo a quantidade de árvores em uma Floresta Aleatória. Ajustar esses valores pode ajudar a reduzir o risco de *overfitting*, além de melhorar o desempenho do modelo. De acordo com (GÉRON, 2017) existem algumas possíveis soluções para evitar este fenômeno:

- Coletar mais dados de treinamento;
- Realizar um tratamento de dados, excluindo os erros e valores discrepantes;
- Fazer a escolha de outro método que se ajuste melhor aos dados, como, por exemplo, trocar um ajuste linear por um ajuste exponencial.

A informação "do melhor método" a ser escolhido pode ser fornecida pela validação cruzada, um sistema de reamostragem dos dados. Sistemas desse tipo retiram amostras de uma base de treinamento de maneira repetida, rodando o modelo sucessivamente, a fim de obter novas informações sobre este e auxiliar na escolha de um método mais adequado (árvore de decisão, floresta aleatória, regressão linear, entre outros), evitando o *overfitting*.

Existem diversos tipos de validação cruzada. Um exemplo é a técnica de validação cruzada k-*fold*, a qual consiste nos seguintes passos, como mostrado na Figura 1.1. Note que o processo do k-*fold* se inicia no Passo 2.

Figura 1.1: Exemplo do passo a passo do funcionamento da técnica k-*fold* de validação cruzada com $k = 5$.



Fonte: Autoria própria.

Passo a passo da Validação Cruzada:

Passo 1 Dividir o conjunto de dados entre treinamento e teste, gerar o modelo inicial com os dados de treinamento, avaliar o modelo com os dados de teste e gerar o valor N de uma determinada métrica.

Passo 2 Dividir os dados de treinamento em k subconjuntos disjuntos de maneira aleatória.

Passo 3 Para $i = 1, \cdots, k$

- 3.1 Separe o subconjunto i como conjunto de teste;
- 3.2 Use os $k - 1$ subconjuntos restantes como conjunto de treinamento;
- 3.3 Gere o modelo a partir dos dados de treinamento;
- 3.4 Avalie o modelo com os dados de teste dessa iteração com a mesma métrica utilizada na Etapa 1, obtendo o valor M_i.

Passo 4 Após todas as iterações, calcule a média M dos valores M_i para as iterações i de 1 até k.

Passo 5 Comparar o valor N da métrica do modelo inicial, com o valor M da métrica obtida pela validação cruzada.

Como o valor M da métrica obtida gerada pela validação não se baseia em uma única divisão em treino e teste, mas utiliza múltiplos subconjuntos, permitindo que cada ponto do conjunto de dados seja utilizado tanto para treinamento quanto para teste, o valor da métrica gerado neste processo fornece uma estimativa mais consistente do desempenho do modelo em comparação com o valor N da métrica do modelo inicial.

Comparar o valor da métrica N, adquirida no Passo 1, com o valor da métrica obtida pela validação cruzada M é um momento essencial para a análise do modelo. Se o valor M for significativamente inferior ao valor inicial N, pode ser uma indicação de *overfitting*, ou seja, o modelo pode ter se ajustado excessivamente aos dados de treinamento, podendo não realizar boas previsões dada uma nova observação. Por outro lado, se o valor de M for próximo a N e o desempenho inicial já tiver sido considerado satisfatório pelo programador, o modelo

inicial pode ser indicado como o escolhido, uma vez que a validação mostrou um desempenho consistente.

Outro aspecto da utilização da validação cruzada é a possibilidade de testar diferentes métodos (como árvore de decisão, floresta aleatória, regressão linear, entre outros). As métricas fornecidas pela validação cruzada permitem comparar qual dos métodos testados pelo programador apresenta o melhor desempenho.

1.1.4 Dados Desbalanceados e Técnica SMOTE

Um conjunto de dados é considerado desbalanceado se suas classes não possuem uma representação equilibrada (BOWYER *et al.*, 2011). Dados sobre treinamento e lesões de atletas podem conter um alto nível de desbalanceamento, no qual a maioria dos registros corresponde a dias sem ocorrência de lesão. Dessa forma, o modelo pode acabar classificando todas as observações como a classe majoritária (sem lesão), resultando em uma alta acurácia. Entretanto, o objetivo do modelo não estaria sendo cumprido, uma vez que modelos desbalanceados geralmente buscam classificar corretamente a classe minoritária, permitindo uma pequena taxa de erro na classe majoritária para que o objetivo seja alcançado (BOWYER *et al.*, 2011). Além disso, para a organização esportiva e a saúde da atleta, o custo de classificá-la erroneamente como "não lesão", quando há chance de lesão, é maior do que o custo de um alarme falso.

Existem diversas técnicas cujo objetivo é tratar dados balanceados. Neste trabalho, é utilizada a Técnica de Sobreamostragem Sintética da Classe Minoritária (BOWYER *et al.*, 2011), do inglês *Synthetic Minority Oversampling Technique* (SMOTE), a qual consiste na criação de dados "sintéticos" da classe minoritária.

A cada iteração do SMOTE, seleciona-se uma observação da classe minoritária e identificam-se os k pontos da classe minoritária com a menor distância. Na sequência, calcula-se o vetor resultante da diferença entre a observação inicial e seu vizinho mais próximo. Esse vetor é multiplicado por um número aleatório entre 0 e 1, gerando a localização de uma nova amostra sintética da classe minoritária. O resultado é um ponto aleatório ao longo do segmento de reta entre a observação e seu vizinho.

Seja P a porcentagem de sobreamostragem desejada, por exemplo, se $P = 300$ e tem-se 10 amostras da classe minoritária, serão criadas três novas amostras sintéticas para cada amostra original, obtendo 40 observações finais da classe minoritária (sendo 30 sintéticas e 10 originais). Seja k a quantidade de vizinhos mais próximos a ser considerada, no geral, $k = 5$. O algoritmo da técnica SMOTE é detalhado a seguir, com base na referência (BOWYER *et al.*, 2011).

Os passos a seguir são realizados para cada amostra x_i da classe minoritária, sendo i de 1 até a quantidade de amostras da classe com menor recorrência.

Passo 1 Selecione uma observação x_i da classe minoritária.

Passo 2 Calcule a distância entre x_i e todas as outras amostras da classe minoritária utilizando a distância euclidiana e identifique os k vizinhos com menor distância.

Passo 3 Escolha aleatoriamente $P/100$ amostras dentre os k vizinhos encontrados.

Passo 4 Para cada vizinho x_v (v de 1 até $P/100$) escolhido no passo anterior, realize os seguintes passos:

4.1 Calcule o vetor resultante da diferença entre x_i e x_v;

4.2 Gere um valor aleatório $\alpha \in (0,1)$;

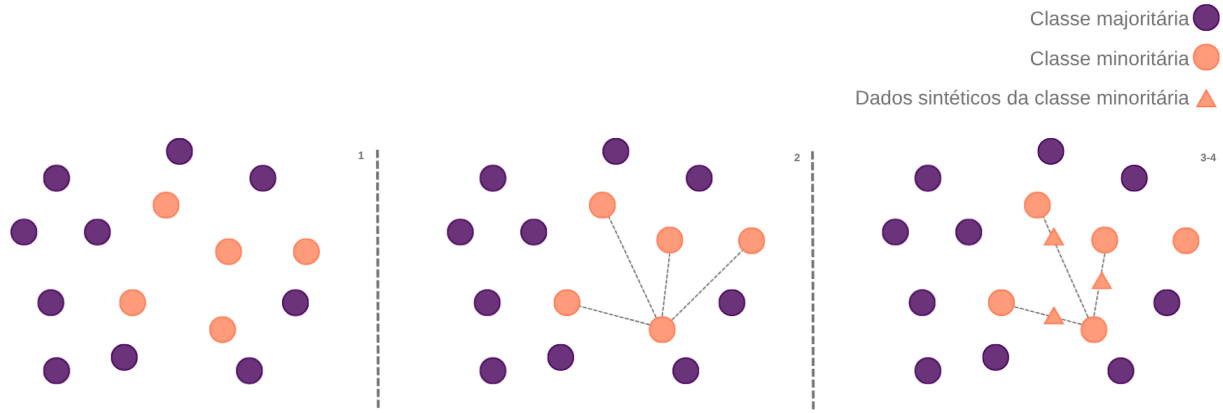
4.3 Crie a localização da amostra sintética x_s através da equação:

$$x_s = x_i + \alpha \cdot (x_v - x_i). \tag{1.6}$$

Passo 5 Encerre o algoritmo se todas as amostras x_i tiverem sido selecionadas. Em seguida, adicione todas as amostras sintéticas x_s ao conjunto de dados.

Na Figura 1.2 é detalhado o desenvolvimento do algoritmo da técnica SMOTE para $P = 300$.

Figura 1.2: Criação de dados sintéticos utilizando o SMOTE.



Fonte: Autoria própria.

1.2 Árvore de Decisão

A árvore de decisão é um método de aprendizado de máquina a qual é utilizada tanto para classificação quanto para regressão de dados. Este método é considerado uma das técnicas mais simples e fáceis de implementar (OKADA; NEVES; SHITSUKA, 2019). O modelo de árvore de decisão possui uma estrutura composta por nós de decisão, que contêm um teste sobre alguma variável de entrada, dividindo os dados de treinamento em subconjuntos menores com variável de saída da mesma classe (TANGIRALA, 2020). Este, por sua vez, gera outro nó de decisão ou um nó folha, que corresponde ao rótulo final, ou seja, a saída do problema (REZENDE, 2003).

Existem diversos algoritmos de árvore de decisão, os quais se diferenciam por características como o número de nós filhos gerados a partir de um nó e os critérios estatísticos utilizados para as divisões. Entre os mais conhecidos estão *Iterative Dichotomiser 3* (ID3), C4.5, *Classification and Regression Trees* (CART) e *Chi-squared Automatic Interaction Detection* (CHAID), cada um com abordagens específicas para a divisão dos dados (SINGH; GIRI, 2014). Neste trabalho, é usado o algoritmo CART, o qual cria divisões estritamente binárias, ou seja, um nó de teste é ligado apenas a outros dois nós e utiliza o índice Gini para medir o nível de impureza do conjunto de dados (OKADA; NEVES; SHITSUKA, 2019), definido pela fórmula

$$Gini(D) = 1 - \sum_{i=1}^{k} (p_i)^2, \tag{1.7}$$

em que k é a quantidade de classes e p_i representa a razão entre a quantidade de amostras da classe i e o total de amostras, ou seja, a frequência relativa da classe i no conjunto de dados D.

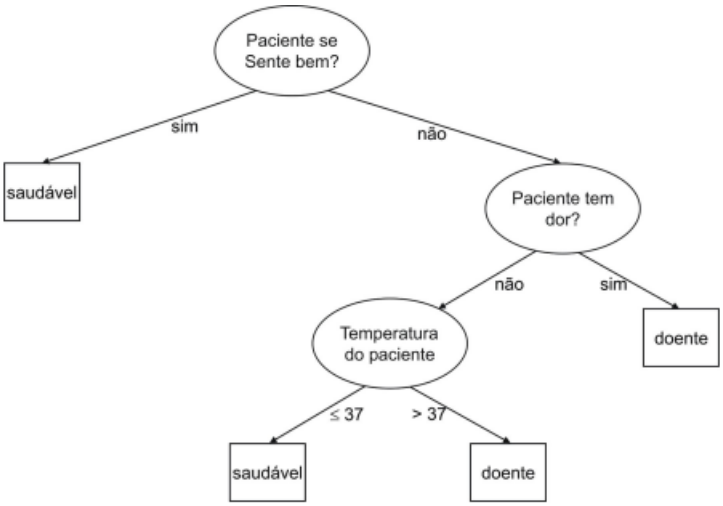
Quanto mais próximo de zero for o índice de Gini de um conjunto, maior é a pureza dos dados. De fato, em uma amostra de dados A totalmente homogênea, com apenas uma classe ($k = 1$), é possível observar que a frequência relativa dessa classe é 1, assim,

$$Gini(A) = 1 - 1^2 = 0. \tag{1.8}$$

Na Figura 1.3 é mostrado um exemplo de uma árvore de decisão para o diagnóstico de um paciente. Na figura, as formas geométricas são as representações dos nós da árvore, sendo

a elipse o nó de decisão (teste) e o retângulo o nó folha (diagnóstico final). As variáveis de entrada são o bem-estar, a presença de dor e a temperatura do paciente em graus, enquanto a saída corresponde ao seu diagnóstico, doente ou saudável.

Figura 1.3: Exemplo de árvore de decisão para diagnóstico de pacientes.



Fonte: (REZENDE, 2003).

Na árvore da Figura 1.3, o primeiro nó de decisão realiza um teste relacionado ao bem-estar do paciente, verificando se ele se sente bem. Se a resposta for positiva, o paciente é considerado saudável. Caso contrário, um novo nó de decisão é gerado, avaliando se o paciente está com dor. Se houver dor, o paciente é considerado doente. Se não houver, outro nó de decisão é criado para analisar sua temperatura corporal. Se a temperatura for menor ou igual a $37°C$, o paciente é classificado como saudável, caso contrário, é considerado doente.

O objetivo da árvore de decisão é identificar, em cada nó, a variável e o atributo que melhor realizam uma divisão a fim de minimizar a heterogeneidade dos rótulos, visando formar os subconjuntos o mais homogêneos possível em cada nó-folha (TANGIRALA, 2020).

Dado um conjunto de dados D e uma variável de entrada v com atributos/instâncias A, B e C, para tomar a decisão sobre qual atributo é utilizado para realizar a partição, inicialmente, é calculado o índice Gini do conjunto D em relação à variável de saída (s) do problema,

$$Gini(s) = 1 - \sum_{i=1}^{k} (p_i)^2, \tag{1.9}$$

em que k é a quantidade de atributos da variável de saída s e p_i a probabilidade de ocorrer cada instância no conjunto D.

A partir disso, calcula-se o índice Gini da divisão para cada atributo. Por exemplo, dada a classe A, o Gini da divisão sobre o atributo A ($v = A$ / $v \neq A$), também conhecido como *Gini Split*, é dado por

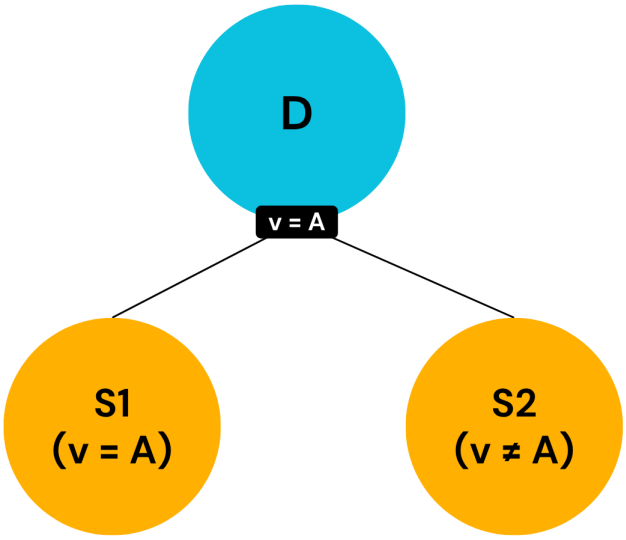
$$GiniSplit(v,A) = \frac{|S_1|}{|D|} Gini(S_1) + \frac{|S_0|}{|D|} Gini(S_0), \tag{1.10}$$

sendo S_1 o subconjunto de D tal que $v = a$ e S_0 o subconjunto de D tal que $v \neq a$, como visto na Figura 1.4. Observe que $\frac{|S_1|}{D}$ é a probabilidade/frequência relativa de S_1 e que

$$Gini(S_1) = 1 - \sum_{i=1}^{k} (p_i)^2, \tag{1.11}$$

em que k é a quantidade de atributos da variável de saída s e p_i a probabilidade de ocorrer cada classe no conjunto S_1.

Figura 1.4: Divisão de uma árvore de decisão sobre a variável v e o atributo A.



Fonte: Autoria própria.

Para determinar qual atributo (A, B ou C) é utilizado na divisão, é escolhido aquele com o menor valor de Gini Split, assim, tem-se o índice Gini da varíavel v,

$$Gini(v) = min\{GiniSplit(v,A),GiniSplit(v,B),GiniSplit(v,C)\}. \tag{1.12}$$

Dessa forma, é possível calcular o ganho de informação da variável de entrada v, dada pela fórmula

$$\Delta Ganho(v) = Gini(s) - Gini(v). \tag{1.13}$$

Caso o conjunto de dados possua outra variável u, cujo ganho de informação seja maior, esta é escolhida para ser utilizada na divisão da árvore de decisão.

1.2.1 Entendendo os Cálculos da Árvore de Decisão

Para exemplificar o cálculo do índice Gini, foi desenvolvido um modelo de árvore de decisão, cujo objetivo é prever a ocorrência ou não de lesão em atletas. Nesse sentido, suponha um conjunto de dados de atletas de futebol de uma liga com campeonatos divididos em 3 divisões: Primeira, Segunda e Terceira, com categorias por idade Júnior (atletas com idades menores que 20 anos), Sênior (atletas com idades entre 20 e 40 anos) e Veterano (atletas com idades maiores que 40 anos), dividido em campeonatos: feminino e masculino. As variáveis de entrada do problema são a divisão do campeonato (d), o sexo do atleta (x) e a categoria à qual o (a) atleta pertence (c), e a saída é a indicação ou não de lesão no atleta (s).

O *dataset* mostrado na Figura 1.5 possui 142 linhas, em que cada linha representa um atleta, e foi dividido em aproximadamente 70% para o conjunto de treinamento e 30% para o conjunto de teste; neste caso, o algoritmo escolheu 102 linhas para treinamento e 40 para teste.

Figura 1.5: Cabeçalho do conjunto de dados da liga de futebol para o modelo.

Divisao	Sexo	Categoria	Lesao
Terceira	Masculino	Senior	Nao
Primeira	Feminino	Senior	Sim
Primeira	Feminino	Senior	Sim
Primeira	Masculino	Senior	Sim
Segunda	Feminino	Senior	Sim
Terceira	Feminino	Senior	Sim
Terceira	Masculino	Senior	Nao
Terceira	Masculino	Senior	Nao
Segunda	Masculino	Veterano	Nao
Primeira	Feminino	Senior	Sim

Fonte: *Software* R.

A Tabela 1.2 representa a quantidade de atletas do conjunto de treinamento que tiveram ou não alguma lesão.

Tabela 1.2: Quantidade de atletas lesionados e não lesionados.

Lesão		Total
Sim	Não	(Treinamento)
48	54	*102*

Assim, é possível calcular o índice Gini do conjunto de dados em relação à variável de saída "Lesão" (s), a qual possui 2 classes "sim" e "não",

$$Gini(s) = 1 - \sum_{i=1}^{2}(p_i)^2 = 1 - \left(\frac{48}{102}\right)^2 - \left(\frac{54}{102}\right)^2 = 0{,}4982699. \tag{1.14}$$

O passo seguinte é calcular o índice Gini de algum possível nó da partição, para exemplificar, é calculado o Gini da variável "Divisão" no atributo "Terceira" (Divisão = Terceira). Com isso, é necessária a quantidade de atletas lesionados e não lesionados do conjunto de treinamento, para os atributos da variável "Divisão", cujas quantidades estão representadas na Tabela 1.3.

Tabela 1.3: Relação de atletas lesionados e não lesionados de cada divisão da liga.

Lesionado / Divisão	Sim	Não
Primeira	20	2
Segunda	17	11
Terceira	11	41

Assim, o cálculo do índice Gini da variável "Divisão" no atributo "Terceira" segue da seguinte forma,

$$GiniSplit(d,T) = \frac{|S_1|}{|D|}Gini(S_1) + \frac{|S_0|}{|D|}Gini(S_0), \tag{1.15}$$

sendo D o conjunto de treinamento, S_1 o subconjunto de D tal que $d = T$ e S_0 o subconjunto tal que $d \neq T$. Observe que em S_0 são considerados os atributos "Primeira divisão" e "Segunda divisão", assim

$$Gini(S_1) = 1 - \sum_{i=1}^{2}(p_i)^2 = 1 - \left(\frac{11}{11+41}\right)^2 - \left(\frac{41}{11+41}\right)^2 = 0{,}3335799, \tag{1.16}$$

$$Gini(S_0) = 1 - \sum_{i=1}^{2}(p_i)^2 = 1 - \left(\frac{20+17}{20+2+17+11}\right)^2 - \left(\frac{2+11}{20+2+17+11}\right)^2 = 0{,}3848. \tag{1.17}$$

Logo,

$$GiniSplit(d,T) = \frac{11+41}{102}0{,}3335799 + \frac{20+2+17+11}{102}0{,}3848 = 0{,}3586878. \tag{1.18}$$

O mesmo procedimento é realizado para as outras duas possíveis separações sobre a variável "Divisão", $(d = P \ / \ d \neq P)$ e $(d = S \ / \ d \neq S)$, sendo P o atributo "Primeira" e S "Segunda",

$$GiniSplit(d,P) = 0{,}3925134 \quad \textbf{e} \quad GiniSplit(d,S) = 0{,}4841585. \tag{1.19}$$

Dessa forma, como o índice Gini da variável "Divisão" no atributo "Terceira" é o menor entre os Gini Splits, tem-se $Gini(d) = 0{,}3586878$. Assim, é possível calcular o ganho de informação sobre a variável "Divisão",

$$\Delta Ganho(d) = Gini(s) - Gini(d) = 0{,}4982699 - 0{,}3586878 = 0{,}1395821. \tag{1.20}$$

As Tabelas 1.4 e 1.5 representam a quantidade de atletas lesionados e não lesionados para cada atributo das variáveis "Sexo" e "Categoria", com base no conjunto de treinamento.

Tabela 1.4: Relação de atletas lesionados e não lesionados de cada sexo.

Sexo \ Lesionado	Sim	Não
Feminino	35	12
Masculino	13	42

Tabela 1.5: Relação de atletas lesionados e não lesionados de cada categoria.

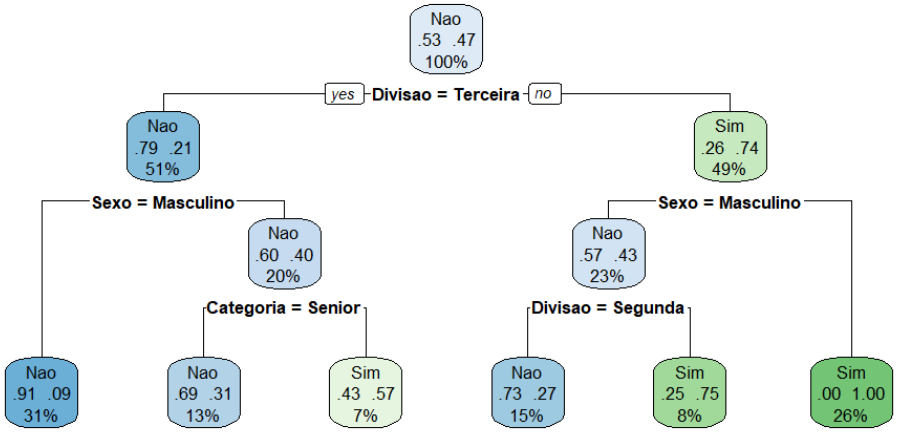
Categoria \ Lesionado	Sim	Não
Sênior	34	42
Veterano	5	6
Júnior	9	6

De forma análoga à equação (1.20), calcula-se o ganho das outras variáveis de entrada, "Sexo" (x) e "Categoria" (c),

$$\Delta Ganho(x) = 0{,}1283985 \quad \textbf{e} \quad \Delta Ganho(c) = 0{,}005774967. \tag{1.21}$$

Como a variável "Divisão" obteve o maior ganho, e dentre seus atributos, "Terceira" foi o que obteve menor índice de Gini, no primeiro nó da árvore é realizado o teste "Divisão = Terceira". Os nós restantes são calculados da mesma maneira, levando em consideração o conjunto de dados do nó atual, em vez de considerar todo o conjunto de treinamento. Na Figura 1.6 é apresentado o modelo final desenvolvido através do *software R*, juntamente com o teste escolhido em cada nó de divisão.

Figura 1.6: Árvore de decisão gerada pelo *software R* com o *dataset* da liga de futebol.



Fonte: *Software* R.

Na seção a seguir é apresentada a teoria do método de aprendizado de máquina Floresta Aleatória, a qual é composta por várias Árvores de Decisão, um método que combina várias árvores pode ser mais eficiente quando comparado a um método de árvore única. Além disso, também é apresentado um exemplo sobre como a Floresta Aleatória pode ser aplicada.

1.3 Floresta Aleatória

A floresta aleatória é um método de aprendizado de máquina a qual é derivada do algoritmo *Classification and Regression Trees* (CART). Desenvolvido por (BREIMAN, 2001), o método utiliza técnicas de aprendizado em conjunto para combinar várias árvores de decisão para obter uma maior precisão em suas previsões.

1.3.1 Aprendizado em Conjunto

O aprendizado em conjunto, também conhecido como *Ensemble Learning*, baseia-se na combinação das decisões de diferentes algoritmos preditivos para desenvolver um modelo mais preciso. Ao contrário do aprendizado comum, que busca construir um único modelo a partir dos dados de treinamento, o aprendizado em conjunto tenta construir um grupo de modelos e combinar suas saídas (ZHOU, 2012).

É possível dividir as técnicas de *Ensemble Learning* em duas categorias: homogêneas e heterogêneas. As técnicas homogêneas são aquelas que possuem a presença de um único método de aprendizado de máquina (como árvore de decisão) e diferentes amostras de dados Z_i do conjunto original Z, $Z_1 \neq Z_2 \neq \cdots \neq Z_n$ (exemplo: *bagging* e *boosting*). As técnicas heterogêneas, por sua vez, utilizam diferentes métodos de aprendizado de máquina e amostras iguais ao do conjunto original $Z_1 = Z_2 = \cdots = Z_n$ (exemplo: *stacking*).

Neste trabalho, é dado enfoque às técnicas heterogêneas, uma vez que o método floresta aleatória apresentado nesta seção utiliza a técnica heterogênea *bagging* para seu desenvolvimento.

Bagging (Bootstrap Aggregating)

O *Bagging (Bootstrap Aggregating)*, traduzido como agregação por *bootstrap*, é uma técnica heterogênea de aprendizado em conjunto. Essa técnica utiliza amostras *bootstrap* em seu funcionamento, as quais são definidas em (ZHOU, 2012) da seguinte maneira:

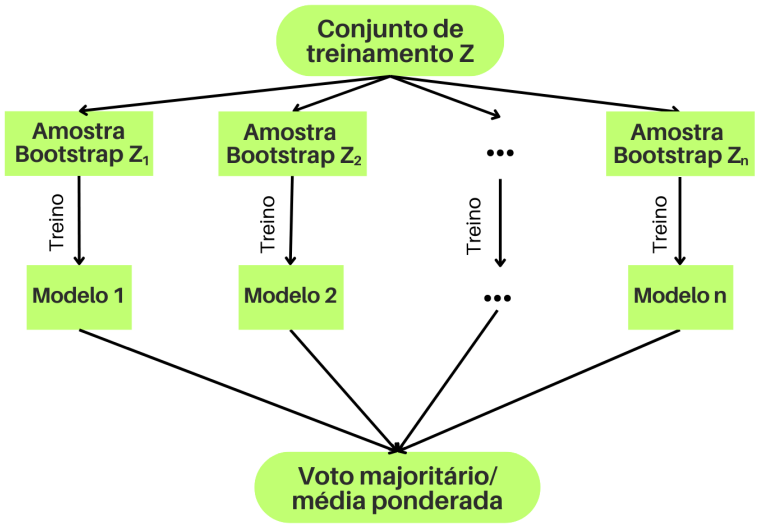
Definição 1.5. *Seja $Z = z_1, z_2, \cdots, z_n$ o conjunto de treinamento, em que $z_i = (x_i, y_i)$, sendo x_i a i-ésima entrada e y_i sua saída, para todo $i \in 1, \cdots, n$. As amostras bootstrap (Z_i) são conjuntos de tamanho n formados por elementos de Z com reposição e de maneira aleatória.*

Por exemplo, em um conjunto de treinamento Z que possui 50 observações, são gerados de maneira aleatória e com reposição, diferentes conjuntos Z_i de tamanho 50 cada.

Na Figura 1.7 é apresentado um diagrama da técnica *bagging*, em que inicialmente são geradas n amostras *bootstraps* Z_i, $i \in \{1, \cdots, n\}$ a partir do conjunto de treinamento Z.

Nessa técnica, o aprendizado dos preditores acontece de forma paralela (ZHOU, 2012). A partir de cada *bootstrap* Z_i, é gerado um modelo preditor utilizando um único método (regressão linear, árvore de decisão, entre outros), ou seja, cada modelo é construído com uma amostragem *bootstrap* diferente. Por exemplo, ao gerar 10 conjuntos *bootstrap*, caso o método escolhido seja árvore de decisão, ao final são obtidas 10 árvores diferentes. Os modelos construídos são independentes; um modelo não influencia o desenvolvimento do outro. Para a predição final da saída de uma nova observação, a técnica utiliza a média das saídas de cada modelo em casos de regressão, enquanto na classificação a predição de cada preditor é combinada por meio do voto majoritário; por exemplo, em um modelo que gerou 10 árvores de decisão, dada uma observação que foi prevista como classe A por 6 árvores e como classe B por 4, o modelo final irá prever esta observação como classe A.

Figura 1.7: Diagrama da técnica de aprendizado em conjunto *bagging*.



Fonte: Autoria própria.

O *bagging* pode melhorar o desempenho de preditores considerados instáveis (ZHOU, 2012), os quais são métodos em que pequenas perturbações nos dados de treinamento podem causar grandes variações nos modelos gerados por estes, como, por exemplo, árvores de decisão, redes neurais, regressão linear, entre outros.

1.3.2 Estrutura da Floresta Aleatória

O método floresta aleatória utiliza duas técnicas de aprendizado em conjunto, o *bagging* de árvores de decisão e o *randomizing*, sendo o último uma técnica em que cada método empregado utiliza-se um conjunto diferente de variáveis em seu conjunto de dados. Inicialmente, a floresta aleatória, em cada nó de cada árvore, não avalia todas as variáveis disponíveis pra saber qual é a "melhor" partição através do índice Gini, como nas árvores de decisão usuais. Neste

momento é realizado o *randomizing*, em que são sorteadas aleatoriamente um conjunto diferente de variáveis, e desse novo conjunto é escolhida a melhor variável para se realizar a partição, esse processo é repetido em cada nó até o fim da árvore (ZHOU, 2012).

Na sequência, apresenta-se o algoritmo de uma floresta aleatória com B árvores, sendo Z o conjunto de dados original de tamanho N, que possui p variáveis de entrada, obtido em (HASTIE; TIBSHIRANI; FRIEDMAN, 2009).

Passo 1 Para $b = 1, \cdots, B$

1.1 Obtenha uma amostra *bootstrap* $Z*$ de tamanho N a partir do conjunto de treinamento.

1.2 Gere uma árvore de decisão T_b a partir da amostra *bootstrap*, repetindo recursivamente os passos a seguir para cada nó da árvore, até o tamanho mínimo do nó n_{min} (quantidade de observações restantes no nó) ser alcançado.

1.2.1 Selecione m variáveis aleatoriamente a partir das p variáveis.

1.2.2 Escolha a melhor variável para a divisão, dentre as m, utilizando o índice Gini.

1.2.3. Divida o nó em dois filhos.

Passo 2 Gere o conjunto de árvores $\{T_b\}_1^B$.

Para fazer a predição de uma nova observação x em variáveis de regressão, é realizado a média das previsões de cada árvore T_b,

$$f_{fa}^B(x) = \frac{1}{B}\sum_{b=1}^{B} T_b(x), \tag{1.22}$$

sendo $T_b(x)$ a previsão da observação x pela árvore T_b. Para problemas de classificação, a predição é feita pela moda das predições em cada árvore, também conhecida por voto majoritário,

$$C_{fa}^B(x) = moda\{C_b(x)\}_1^B, \tag{1.23}$$

sendo $C_b(x)$ a classe da observação x predita pela árvore T_b.

Outra diferença entre problemas de classificação e regressão está nos valores padrões para a quantidade de variáveis m selecionadas aleatoriamente e o tamanho mínimo do nó. Para classificação, $m = \lfloor\sqrt{p}\rfloor$ e o tamanho mínimo do nó é 1, enquanto para regressão, $m = \lfloor\frac{p}{3}\rfloor$ e o tamanho mínimo do nó recomendado é 5. Na prática, os melhores valores para esses parâmetros dependem de fato do problema em questão, com isso devem ser tratados como parâmetros de ajuste (HASTIE; TIBSHIRANI; FRIEDMAN, 2009).

Erro das Amostras "*Out-of-bag*" (OOB)

Na técnica *bagging*, quando é escolhida uma amostra *bootstrap*, as observações são selecionadas com reposição. Em média, apenas cerca de 2/3 das observações são escolhidas para compor uma amostra *bootstrap* (JAMES *et al.*, 2021). Como resultado, cerca de 1/3 das observações do conjunto original de dados não são incluídas na amostra usada para treinar determinada árvore. Essas observações deixadas de fora possuem o nome *out-of-bag* (OOB). Para cada observação OOB, é possível realizar a previsão de sua saída, em cada árvore na qual a observação OOB não é incluída na amostra *bootstrap*.

Dessa maneira, é gerado cerca de $B/3$ previsões para a i-ésima observação, sendo B a quantidade de árvores da floresta. A predição final da i-ésima observação é dada pela moda das

predições, caso o modelo seja de classificação, ou pela média, em casos de regressão. Assim, é possível calcular o erro utilizando as métricas padrões em cada tipo de modelo. Como a saída prevista de cada observação é gerada utilizando apenas as árvores em que não foram utilizadas aquela observação para seu treinamento, o erro OOB é considerado uma métrica válida para avaliação do modelo (JAMES *et al.*, 2021).

Na sequência, apresenta-se o algoritmo para se obter o erro *out-of-bag* em uma floresta aleatória:

Passo 1 Selecione uma observação x_i, $i \in \{1,...,n\}$, sendo n o tamanho do conjunto de dados.

Passo 2 Identifique todas as árvores da floresta aleatória em que x_i não foi utilizada na amostra *bootstrap* de treinamento dessa árvore.

Passo 3 Utilize essas árvores para prever a variável de saída da observação x_i.

Passo 4 Se o problema for de classificação, a predição final $\hat{y}_i$ de x_i é obtida pela moda das predições obtidas no item anterior. Caso seja de regressão, a predição é obtida pela média das predições.

Passo 5 Calcule o erro utilizando as métricas usuais de cada tipo de modelo.

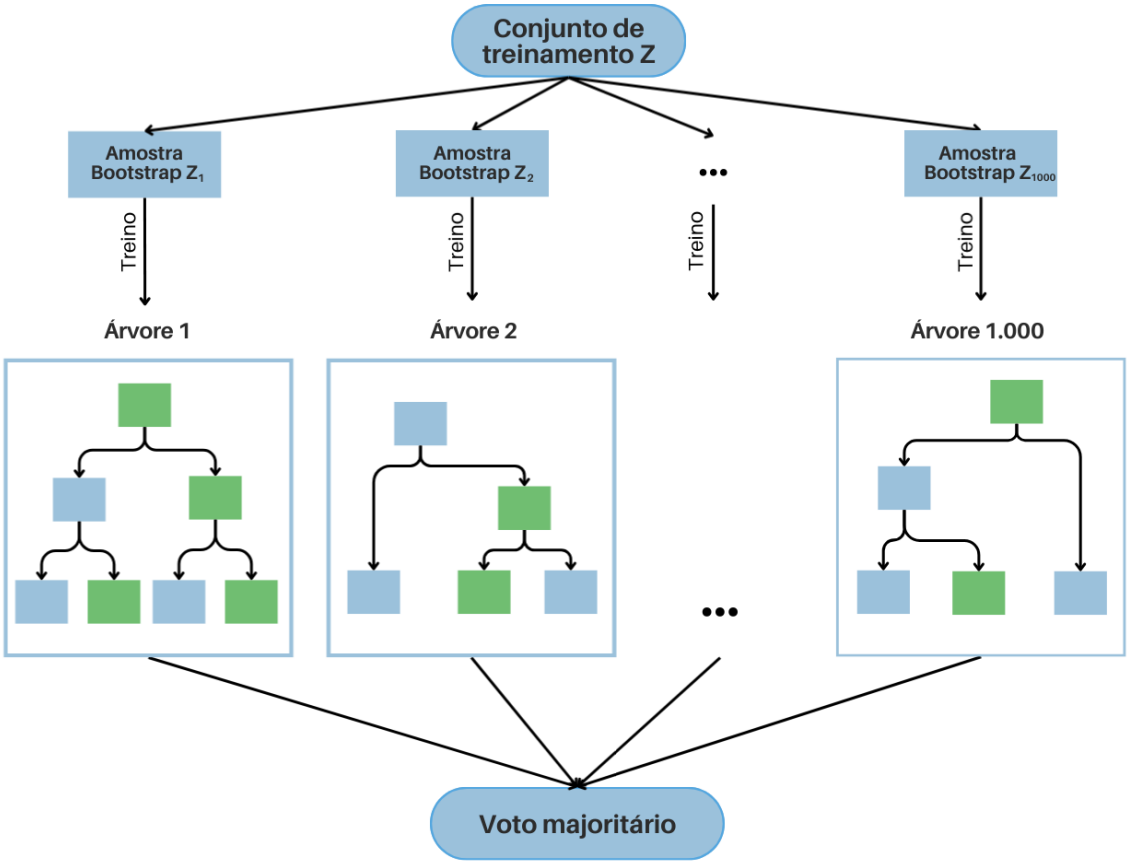
Estimativas "*out-of-bag*" ajudam a evitar a necessidade de um conjunto de dados de validação independente. A cada iteração, à medida que o número de árvores aumenta no método de floresta aleatória, o erro OOB pode ser calculado. Dessa forma, alcançada a estabilidade do erro OOB, o treinamento do modelo pode ser finalizado (HASTIE; TIBSHIRANI; FRIEDMAN, 2009).

1.3.3 Exemplo de Floresta Aleatória

Para ilustrar a construção do modelo de Floresta Aleatória, considere o conjunto de dados da liga de futebol presente na Figura 1.5. O *dataset* possui 142 observações e, como variáveis de entrada, "Divisão", "Sexo" e "Categoria", e como saída, a indicação de lesão (Sim ou Não). Foi utilizado 70% do conjunto de dados para treinamento e 30% para teste, e foi escolhida a quantidade de 1.000 árvores para o desenvolvimento da floresta.

Na Figura 1.8, é mostrada a estrutura da floresta aleatória. Inicialmente, o conjunto de dados de treinamento é dividido em 1.000 amostras *bootstrap*. Para cada amostra, é treinada e desenvolvida uma árvore de decisão. Dada uma observação, sua saída é o voto majoritário (moda) das saídas de cada árvore.

Figura 1.8: Estrutura floresta aleatória.



Fonte: Autoria própria.

A partir do modelo gerado, é possível calcular a predição de lesão ou não lesão de acordo com os dados de uma nova atleta. Na Tabela 1.6 é ilustrada a predição da indicação de lesão realizada pelo modelo para três atletas diferentes.

Tabela 1.6: Predição da indicação de lesão de novas atletas pelo modelo Floresta Aleatória

Caso	Variáveis de entrada			Variável de Saída
	Sexo	Divisão	Categoria	Indicação de Lesão (Sim ou Não)
Atleta 1	Feminino	Primeira	Sênior	Sim
Atleta 2	Feminino	Terceira	Sênior	Não
Atleta 3	Feminino	Segunda	Veterano	Sim

Na seção a seguir, é introduzido o conceito de redes neurais artificiais, apresentando o surgimento dos *perceptrons* de camada única e a motivação para a criação dos *perceptrons* de múltiplas camadas. As redes neurais, em geral, conseguem modelar interações mais complexas entre as variáveis de entrada e a variável de saída quando comparadas à Árvore de Decisão e à Floresta Aleatória.

1.4 Redes Neurais Artificiais

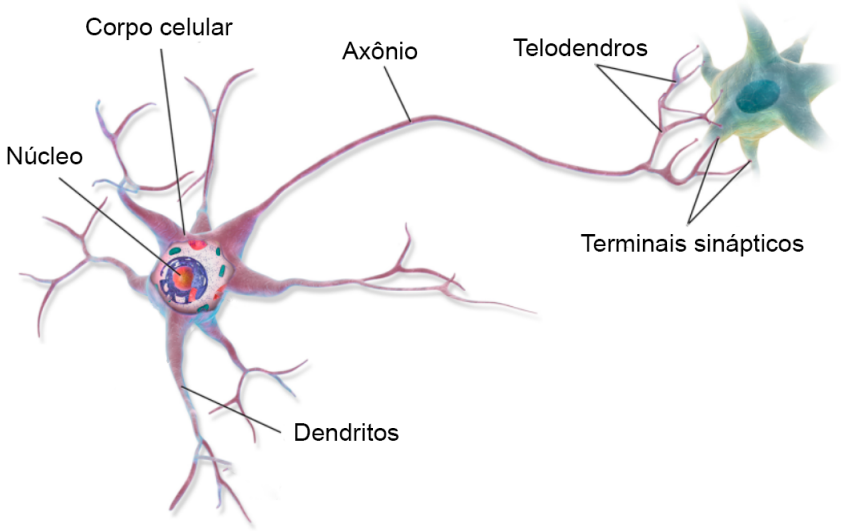
Em 1943, o neurofisiologista Warren McCulloch e o matemático Walter Pitts observaram o funcionamento de neurônios biológicos em animais e, a partir dessa observação, buscaram

apresentar um modelo matemático e computacional simplificado sobre como os neurônios do sistema nervoso processam informações, utilizando proposições lógicas. Dessa forma, McCulloch e Pitts introduziram o conceito de Redes Neurais Artificiais (RNAs) em seu artigo "*A Logical Calculus of Ideas Immanent in Nervous Activity*" (Um Cálculo Lógico de Ideias Iminentes na Atividade Nervosa) (MCCULLOCH; PITTS, 1943). No final da década de 1980, houve um aumento nos estudos sobre o conexionismo, uma abordagem da inteligência artificial que busca modelar processos mentais. Dessa forma, novas arquiteturas e técnicas de treinamento foram desenvolvidas nesse período (GÉRON, 2017).

Posteriormente, surgiram novos métodos, como o *Boosting* e as Florestas Aleatórias. Nesse contexto, considerando que, no período em questão as redes neurais exigiam um alto grau de ajuste manual, enquanto as novas tecnologias eram automáticas, as redes caíram em desuso pelos pesquisadores (JAMES *et al.*, 2021). Quase três décadas depois, em 2010, as redes neurais ressurgiram com o título de *deep learning* (aprendizado profundo). Pesquisadores continuaram os estudos buscando desenvolver novas arquiteturas e tecnologias adicionais, explorando problemas como classificação de imagens e vídeos, reconhecimento de voz, recomendação de conteúdos, entre outros. Acredita-se que o principal motivo do grande retorno e sucesso das redes neurais atualmente é a alta disponibilidade de conjuntos de dados de treinamento, os quais surgiram devido ao processo de digitalização da ciência e indústria (JAMES *et al.*, 2021).

Na Figura 1.9 é apresentada uma representação simplificada de um neurônio biológico, que é encontrado principalmente nos córtices cerebrais de animais. O neurônio é composto por um corpo celular que contém o núcleo, responsável por armazenar as informações genéticas. Nesse local, ocorrem reações químicas e elétricas que determinam se o neurônio irá ou não disparar a informação recebida. Nas ramificações do corpo celular, estão presentes os dendritos. Eles são responsáveis por receber informações químicas e impulsos de outros neurônios. Partindo do corpo celular, há o axônio, a porta de saída do neurônio, sendo responsável por transmitir as informações para outras células. Em suas extremidades, o axônio se divide em ramificações chamadas telodendros, os quais possuem em suas pontas os terminais sinápticos, também conhecidos como sinapses, estruturas que se conectam aos dendritos (ou corpo celular) de outros neurônios (GÉRON, 2017). É através das sinapses que os neurônios recebem curtos impulsos elétricos, também chamados de sinais, de outros neurônios.

Figura 1.9: Ilustração de um neurônio biológico. Adaptado de (GÉRON, 2017).

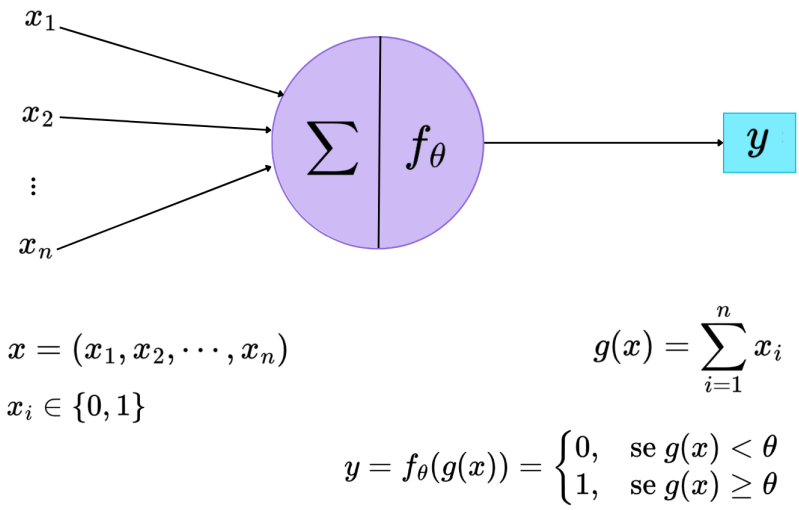


Fonte: (GÉRON, 2017).

Dessa forma, Warren McCulloch e Walter Pitts propuseram um modelo matemático simplificado do funcionamento de um neurônio biológico: o neurônio artificial. Este possui n entradas

binárias x_i, $i \in \{1, \cdots, n\}$ e uma saída y também binária, que é ativada pelo neurônio quando mais do que uma certa quantidade θ de entradas ultrapassam determinado limiar. Neste modelo, todas as entradas possuem o mesmo peso. Na Figura 1.10 é mostrada a representação matemática do neurônio artificial, em que as entradas binárias x_i são somadas por meio da função g e o resultado passa pela função de ativação f_θ, sendo ativada, ou seja, igual a 1, se $g(x) \geqslant \theta$, ou igual a 0 caso contrário.

Figura 1.10: Representação matemática de um neurônio artificial proposto por McCulloch e Pitts.



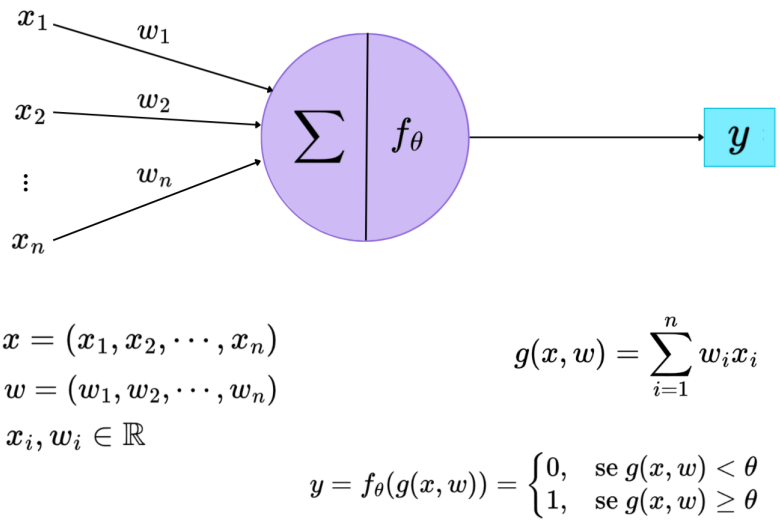
Fonte: Autoria própria.

Apesar de sua simplicidade, McCulloch e Pitts mostraram que é possível computar qualquer proposição lógica construindo uma rede de neurônios artificiais (GÉRON, 2017).

1.4.1 Perceptrons de Camada Única

Inspirado no trabalho de McCulloch e Pitts, aproximadamente 14 anos depois, Frank Rosenblatt propôs o *perceptron* (ROSENBLATT, 1958), uma arquitetura de rede neural artificial baseada em um neurônio artificial modificado, chamado de unidade de limiar lógico, do inglês *threshold logic unit* (TLU). Esse novo neurônio possui entradas e saída sendo números (em vez de valores binários) e cada entrada x_i está associada a um peso w_i, para $i \in \{1, \cdots, n\}$ (GÉRON, 2017). O *perceptron* calcula a soma ponderada de suas n entradas ($w_1 x_1 + w_2 x_2 + \cdots + w_n x_n$) e, então, aplica uma função degrau f_θ a esse resultado para obter sua saída y. Na Figura 1.11 é mostrada a representação matemática de um *perceptron*.

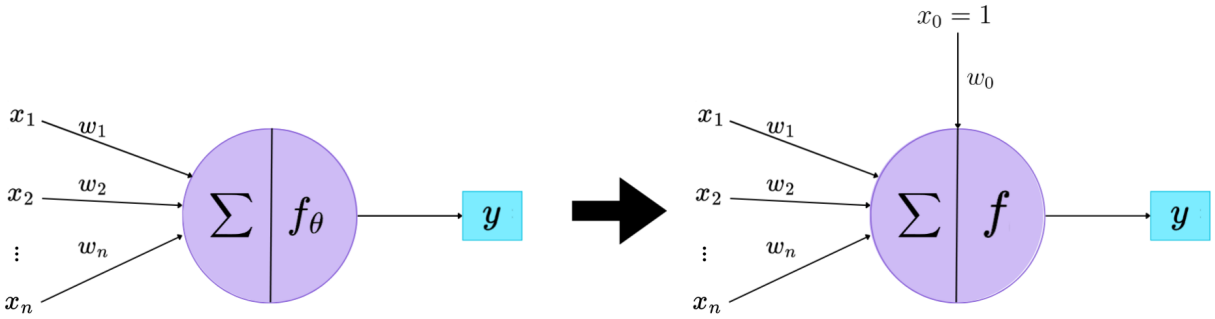
Figura 1.11: Representação matemática de um *perceptron*.



Fonte: Autoria própria.

Pensando no custo computacional, é possível introduzir um valor w_0 conhecido como viés, no lugar do valor de limiar θ. Na Figura 1.12 é mostrado que o limiar pode ser interpretado como o peso de conexão entre a unidade de saída e um sinal de entrada "fictício" x_0, que é sempre igual a 1 (JANG; SUN; MIZUTANI, 1997).

Figura 1.12: Introdução do peso de conexão de viés em um Perceptron.



Fonte: Autoria própria.

Dessa forma, a saída do *perceptron* pode ser reescrita como:

$$y = f(g(x,w)) = f\left(w_1x_1 + w_2x_2 + \cdots + w_nx_n + w_0\right), \tag{1.24}$$

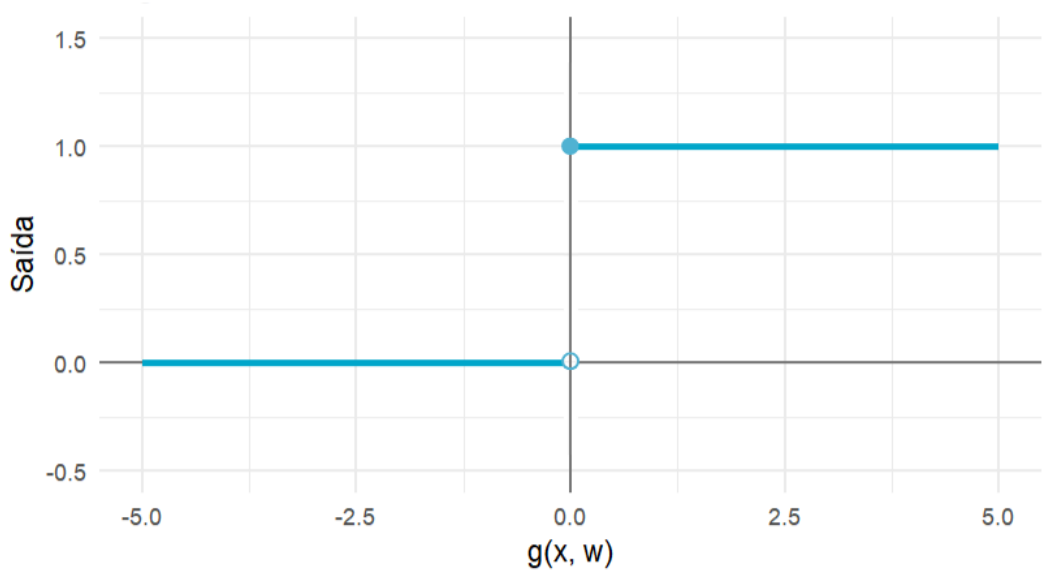
sendo f a função de ativação, $w_1, x_2, \cdots, w_n$ os pesos, x_1,$x_2, \cdots, x_n$ as entradas e w_0 o peso de viés.

A função degrau unitário é uma das funções de ativação mais comuns em redes neurais, podendo também ser utilizada a função sinal. As definições dessas funções são apresentadas a seguir:

$$\text{degrau unitário}(g(x,w)) = \begin{cases} 0, & \text{se } g(x,w) < 0 \\ 1, & \text{se } g(x,w) \geqslant 0, \end{cases} \qquad \text{sinal}(g(x,w)) = \begin{cases} -1, & \text{se } g(x,w) < 0 \\ 0, & \text{se } g(x,w) = 0 \\ +1, & \text{se } g(x,w) > 0 \end{cases}.$$

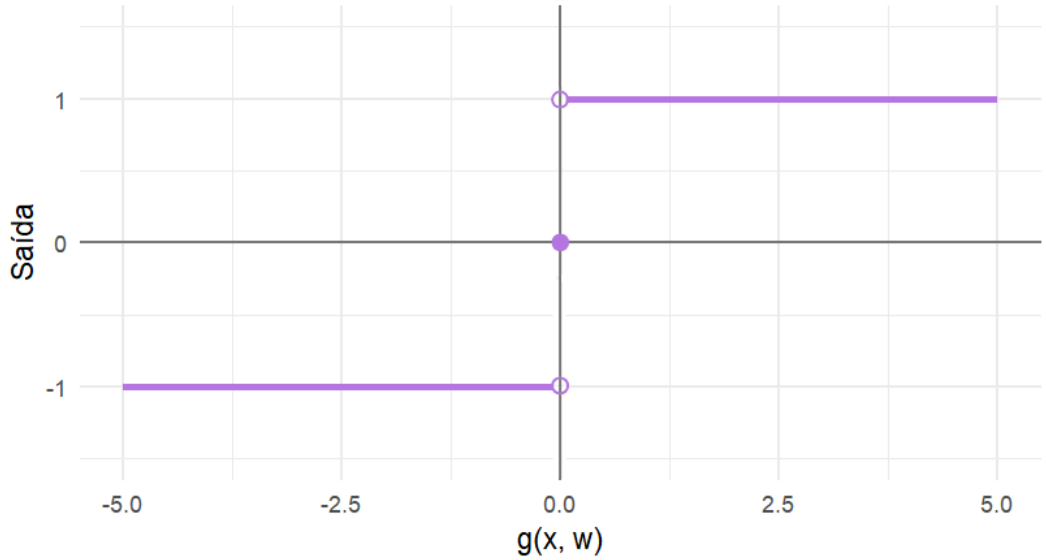
Nas Figuras 1.13 e 1.14 são ilustrados os gráficos das funções degrau unitário e sinal, respectivamente.

Figura 1.13: Gráfico da função degrau unitário.



Fonte: Autoria própria.

Figura 1.14: Gráfico da função sinal.



Fonte: Autoria própria.

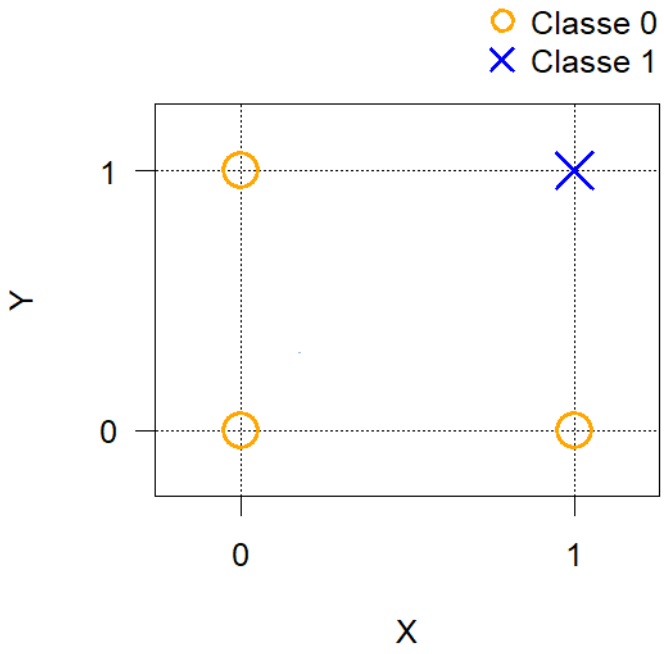
O *perceptron* representado na Figura 1.11 é conhecido como *perceptron* de camada única (*single-layer perceptron*), uma vez que possui apenas uma camada de neurônios. O *perceptron* de camada única possui a capacidade de modelar funções lógicas simples, como as funções AND e OR, pois esses casos são problemas linearmente separáveis, ou seja, suas saídas podem ser separadas por uma reta. Considere a função lógica AND, cuja tabela verdade está apresentada na Tabela 1.7.

Tabela 1.7: Tabela verdade da função lógica AND.

Entrada x_1	Entrada x_2	Saída y
0	0	0
0	1	0
1	0	0
1	1	1

O problema AND é linearmente separável; isso pode ser observado na Figura 1.15, onde é ilustrado uma representação gráfica da função lógica AND. Nesta figura é possível observar que a saída $y = 1$ pode ser separada das saídas $y = 0$ por meio de uma reta.

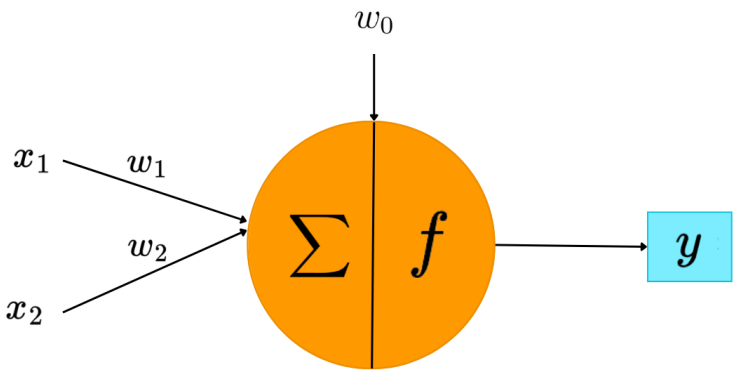
Figura 1.15: Representação gráfica da função lógica AND.



Fonte: Autoria própria.

Na Figura 1.15, os símbolos "x" em azul representam as saídas $y = 1$, enquanto o símbolo "o" em laranja representa a saída $y = 0$. O *perceptron* de camada única que modela a função AND pode ser representado pela Figura 1.16, onde w_1 e w_2 são os pesos das entradas x_1 e x_2, respectivamente, e w_0 é o viés. A função de ativação utilizada é a degrau unitário.

Figura 1.16: Arquitetura do *perceptron* de camada única que modela a função lógica AND.



Fonte: Autoria própria.

Matematicamente, modelar a função lógica AND utilizando um *Perceptron* de camada única com a função de ativação degrau unitário, consiste em encontrar os pesos w_1, w_2 e o viés w_0 que satisfazem as seguintes inequações:

$$
\begin{aligned}
0 \times w_1 + 0 \times w_2 + w_0 < 0, &\iff w_0 < 0,\\
0 \times w_1 + 1 \times w_2 + w_0 < 0, &\iff w_0 < -w_2,\\
1 \times w_1 + 0 \times w_2 + w_0 < 0, &\iff w_0 < -w_1,\\
1 \times w_1 + 1 \times w_2 + w_0 \geqslant 0, &\iff w_0 \geqslant w_1 + w_2.
\end{aligned}
$$

Uma solução possível para essas inequações é $w_1 = 1$, $w_2 = 1$ e $w_0 = -1{,}5$. Dessa forma, uma possível função que modela a função lógica AND pode ser dada por:

$$y = f(1 \times x_1 + 1 \times x_2 - 1{,}5),$$

sendo f a função degrau unitário. Com essa função, é possível obter as saídas da função lógica AND, conforme apresentado na Tabela 1.7.

1.5 Perceptrons de Múltiplas Camadas

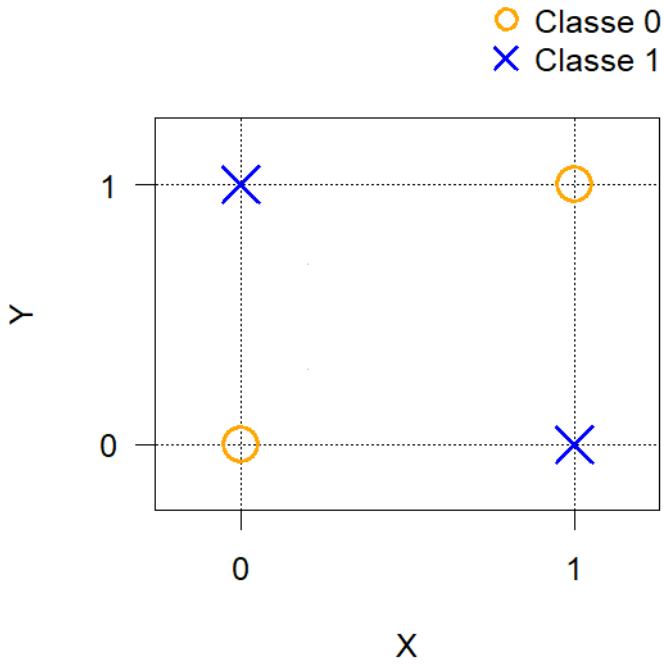
Os perceptrons de camada única geraram um grande interesse na comunidade científica na construção de sistemas inteligentes de autoaprendizagem (JANG; SUN; MIZUTANI, 1997). Entretanto, após a publicação do livro "*Perceptrons*" de Minsky e Papert em 1969, esse entusiasmo foi quebrado (MINSKY; PAPERT, 1969). Neste livro, os autores mostraram que *perceptrons* de camada única podem ser utilizados apenas em problemas simplificados. Um dos resultados apresentados no livro mostra que um *perceptron* de camada única não consegue representar a simples função lógica "ou exclusivo" (*exclusive-OR*), também conhecida como XOR, cuja tabela verdade está apresentada na Tabela 1.8.

Tabela 1.8: Tabela verdade da função lógica XOR.

Entrada x_1	Entrada x_2	Saída y
0	0	0
0	1	1
1	0	1
1	1	0

Dado um vetor de entradas binárias, a tarefa da função XOR é classificar como 0 se o vetor possui uma quantidade par de entradas igual a 1, caso contrário a saída é classificada como 0 (JANG; SUN; MIZUTANI, 1997). Esse problema não é linearmente separável, como ilustrado na Figura 1.17, onde observa-se que não é possível separar as saídas $y = 1$ das saídas $y = 0$ utilizando uma reta.

Figura 1.17: Representação gráfica da função lógica XOR.



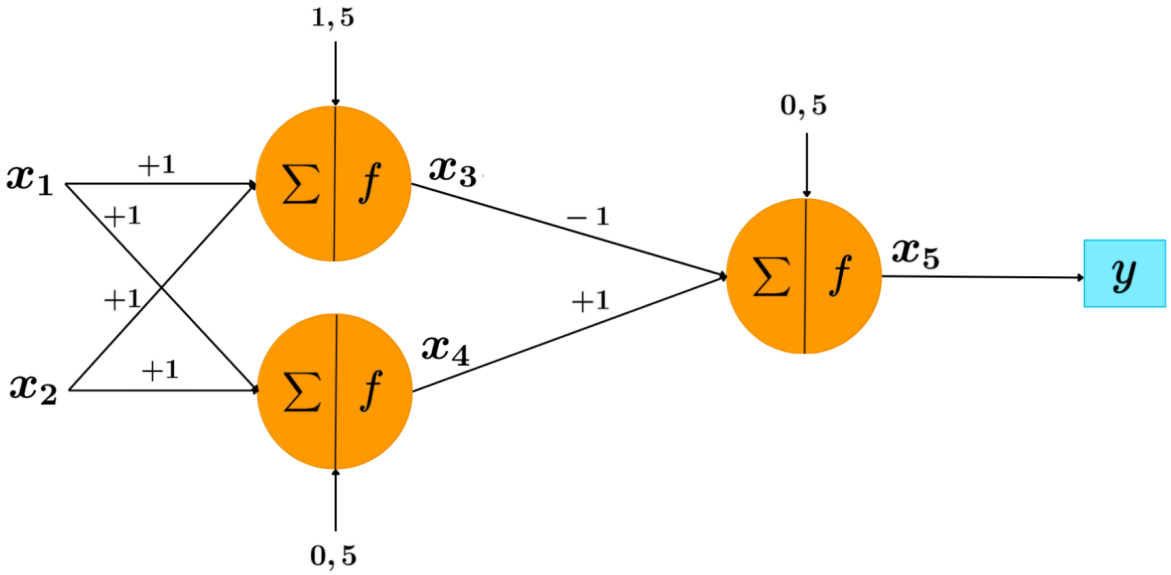
Fonte: Autoria própria.

Matematicamente, modelar a função lógica XOR utilizando um *perceptron* de camada única com a função de ativação degrau unitário, consiste em encontrar os pesos w_1, w_2 e o viés w_0 que satisfazem as seguintes inequações:

$$
\begin{aligned}
0 \times w_1 + 0 \times w_2 + w_0 < 0, &\iff w_0 < 0, \\
0 \times w_1 + 1 \times w_2 + w_0 \geqslant 0, &\iff w_0 \geqslant -w_2, \\
1 \times w_1 + 0 \times w_2 + w_0 \geqslant 0, &\iff w_0 \geqslant -w_1, \\
1 \times w_1 + 1 \times w_2 + w_0 < 0, &\iff w_0 < w_1 + w_2.
\end{aligned}
$$

O sistema de inequações anterior não possui solução, dessa forma não é possível modelar a função lógica XOR utilizando um *perceptron* de camada única. Assim, surge a necessidade de arquiteturas de redes neurais artificiais mais complexas, como o *Perceptron* de Múltiplas Camadas, em inglês *Multi-Layer Perceptron* (MLP). Essa arquitetura conta com mais de uma camada, em que a saída de um neurônio é conectada à entrada de outro neurônio, permitindo a modelagem de problemas mais complexos. Na Figura 1.18 é ilustrada uma arquitetura de um MLP de duas camadas capaz de modelar a função lógica XOR.

Figura 1.18: Arquitetura de um MLP que modela a função lógica XOR.



Fonte: Autoria própria.

O MLP da Figura 1.18 possui duas camadas, sendo a primeira camada composta por dois neurônios e a segunda camada composta por um neurônio. Neste MLP é utilizado o degrau unitário como função de ativação em todos os neurônios. Os pesos e viéses das conexões entre os neurônios neste caso são indicados na Figura 1.18.

A saída do MLP pode ser obtida por meio das seguintes expressões matemáticas:

$$\begin{aligned} x_3 &= f(1 \times x_1 + 1 \times x_2 - 1{,}5), \\ x_4 &= f(1 \times x_1 + 1 \times x_2 - 0{,}5), \\ y = x_5 &= f((-1) \times x_3 + 1 \times x_4 - 0{,}5), \end{aligned}$$

sendo f a função degrau unitário, x_1 e x_2 as entradas, x_3 e x_4 as saídas da primeira camada e y (ou x_5) a saída final do MLP.

Os *perceptrons* de múltiplas camadas podem resolver problemas não linearmente separáveis, como o problema XOR, possuindo uma maior versatilidade em comparação ao *perceptron* de apenas uma camada (JANG; SUN; MIZUTANI, 1997).

No capítulo a seguir, é introduzida a teoria dos conjuntos fuzzy, a qual é utilizada na construção dos sistemas de inferência fuzzy utilizados neste trabalho.

Capítulo 2

Teoria dos Conjuntos Fuzzy

"A computação suave é uma abordagem emergente de computação que traça um paralelo com a notável habilidade da mente humana de raciocinar e aprender em um ambiente de incerteza e imprecisão." (Lotfi A. Zadeh, 1992)

Ao modelar problemas do mundo real, conceitos subjetivos podem surgir; como, por exemplo, a velocidade de um atleta. Se o conjunto dos atletas considerados rápidos for atletas que correm a uma velocidade de 10 km/h, um atleta A_1 que corre a uma velocidade de 9,9 km/h não seria considerado rápido. O ser humano possui a capacidade de compreender que o atleta A_1 pode ser considerado razoavelmente rápido, e essa informação pode ser de grande importância na modelagem de um problema que envolve a velocidade de atletas, a qual poderia ser desconsiderada em modelos baseados na teoria clássica dos conjuntos. Uma maneira de considerar tal informação é considerar atletas rápidos com mais ou menos intensidade, ou seja, existem atletas que pertenceriam mais à classe dos rápidos do que outros (BANDO, 2002). Dessa forma surge a teoria dos conjuntos fuzzy, introduzida por Lotfi A. Zadeh em 1965 (ZADEH, 1965), esta foi desenvolvida para lidar com a imprecisão e a incerteza em modelos matemáticos.

Os conjuntos fuzzy surgem como uma extensão dos conjuntos clássicos, uma vez que a teoria fuzzy considera termos linguísticos como "aproximadamente", "em torno de", dentre outros (JAFELICE; BARROS; BASSANEZI, 2023). Dessa forma, a teoria dos conjuntos fuzzy surge como uma abordagem com grande potencial em problemas de modelagem que lidam com algum tipo de imprecisão, buscando uma aproximação mais realista do modelo em questão.

2.1 Conjuntos Fuzzy

A ideia de conjunto fuzzy serve como uma base para a criação de uma teoria que pode ser considerada semelhante à teoria dos conjuntos clássicos (ZADEH, 1965). Entretanto, a teoria fuzzy pode ser vista como mais abrangente e flexível do que a dos conjuntos clássicos, os quais são definidos por Zadeh através de sua função característica. Neste capítulo, são apresentadas definições referentes à teoria dos conjuntos fuzzy, as quais são baseadas em (JAFELICE; BARROS; BASSANEZI, 2023).

Definição 2.1. *Um subconjunto clássico A de um universo U é caracterizado por sua função característica $\chi_A(x) : U \to \{0,1\}$, definida por partes como:*

$$\chi_A(x) = \begin{cases} 1, & se\ x \in A \\ 0, & se\ x \notin A \end{cases}.$$

Definição 2.2. *Um subconjunto fuzzy A de um universo U é definido em termos de uma função de pertinência μ_A que a cada elemento x de U associa um número $\mu_A(x)$ entre zero e um, chamado de grau de pertinência de x a A. Assim,*

$$\mu_A : U \to [0, 1]. \tag{2.1}$$

Os valores $\mu_A(x) = 1$ e $\mu_A(x) = 0$ indicam, respectivamente, a pertinência plena e a não pertinência do elemento x a A. Também é possível representar um subconjunto fuzzy como um conjunto clássico de pares ordenados da seguinte maneira:

$$G = \{(x, \mu_A(x)) \mid x \in U\},$$

sendo G o gráfico da função de pertinência $\mu_A(x)$.

Destaca-se que um subconjunto clássico A de U é um caso particular de conjunto fuzzy, sendo a função característica a sua função de pertinência, ou seja,

$$\mu_A : U \to \{0, 1\}.$$

O exemplo a seguir ilustra como conjuntos fuzzy são capazes de modelar incertezas linguísticas em um contexto matemático.

Exemplo 2.1. *Seja U o conjunto das velocidades alcançadas por atletas de futebol e A o subconjunto das velocidades consideradas de intensidade muito alta. Considere uma velocidade "muito alta" como sendo aquelas iguais ou superiores a* 19,8 *km/h, como citado em (FAUDE; KOCH; MEYER, 2012).*

Na teoria dos conjuntos clássicos, velocidades de 19 *km/h e* 10 *km/h alcançadas por jogadoras não seriam consideradas "muito altas", logo não pertenceriam ao subconjunto A. Entretanto, tomando A como um conjunto fuzzy, as velocidades de* 19 *km/h e* 10 *km/h poderiam pertencer ao subconjunto A, porém com graus de pertinência (μ_A) diferentes, sendo μ_A a função de pertinência em relação ao conjunto das velocidades consideradas "muito altas". Como* 19 *km/h está mais perto de ser considerada uma velocidade "muito alta" do que* 10 *km/h, observa-se que*

$$\mu_A(19) > \mu_A(10).$$

Como velocidades consideradas muito altas são aquelas iguais ou superiores a 19,8 *km/h, tem-se*

$$\mu_A(x) = 1, \quad \forall x \geqslant 19{,}8,$$

sendo x a velocidade alcançada por uma jogadora de futebol em km/h.

2.2 Operações entre Conjuntos Fuzzy

Nesta seção, para definir as operações entre conjuntos fuzzy, são apresentadas operações realizadas em conjuntos clássicos e suas respectivas funções características.

Sejam A e B subconjuntos clássicos de um universo U, representados pelas funções características χ_A e χ_B, respectivamente. Os conjuntos

$$\begin{gathered} A \cup B = \{x \in U \mid x \in A \text{ ou } x \in B\} \\ A \cap B = \{x \in U \mid x \in A \text{ e } x \in B\} \\ A^c = \{x \in U \mid x \notin A\}. \end{gathered}$$

podem ser representados pelas seguintes funções características,

$$\chi_{A\cup B}(x) = \max\{\chi_A(x), \chi_B(x)\}$$
$$\chi_{A\cap B}(x) = \min\{\chi_A(x), \chi_B(x)\}$$
$$\chi_{A^c}(x) = 1 - \chi_A(x).$$

De fato, observe que, dado $x \in U$:

$$x \in A \cup B \iff \chi_{A\cup B}(x) = 1 \iff \chi_A(x) = 1 \text{ ou } \chi_B(x) = 1 \iff \{x \in A \text{ ou } x \in B\}.$$

O mesmo acontece para as operações de interseção e complemento:

$$x \in A \cap B \iff \chi_{A\cap B}(x) = 1 \iff \chi_A(x) = 1 \text{ e } \chi_B(x) = 1 \iff \{x \in A \text{ e } x \in B\}$$

$$x \in A^c \iff \chi_{A^c}(x) = 1 \iff \chi_A(x) = 0 \iff x \notin A.$$

Definição 2.3. *Sejam A e B subconjuntos fuzzy de um universo U. Os subconjuntos fuzzy $A \cup B$, $A \cap B$ e A^c são definidos através das seguintes funções de pertinência:*

$$\mu_{A\cup B}(x) = \max\{\mu_A(x), \mu_B(x)\}$$
$$\mu_{A\cap B}(x) = \min\{\mu_A(x), \mu_B(x)\}$$
$$\mu_{A^c}(x) = 1 - \mu_A(x).$$

Uma das principais diferenças entre a teoria clássica e a teoria fuzzy é encontrada na operação do complemento. Dado A um subconjunto clássico de U, então

$$\mu_A(x) = 1 \text{ e } \mu_{A^c}(x) = 0, \text{ então } x \in A,$$

ou

$$\mu_A(x) = 0 \text{ e } \mu_{A^c}(x) = 1, \text{ então } x \notin A.$$

Ou seja, $x \in A$ ou $x \notin A$. Entretanto, caso A seja um conjunto fuzzy, essa dicotomia não ocorre necessariamente, da mesma forma que pode não acontecer

$$A \cap A^c = \varnothing \text{ e } A \cup A^c = U.$$

O exemplo a seguir mostra uma situação em que ocorre tal fato.

Exemplo 2.2. *Suponha U o conjunto universo composto por atletas de um time, identificados pelos números 1, 2, 3, 4 e 5. Sejam A e B conjuntos fuzzy que representam os atletas com fadiga e sobrecarga, respectivamente. Na Tabela 2.1 é apresentado o grau de pertinência de cada atleta em relação a união, interseção e o complemento dos conjuntos A e B.*

Tabela 2.1: Grau de pertinência de cada atleta aos conjuntos A e B e suas operações.

Atleta	Fadiga(μ_A)	Sobrecarga(μ_B)	$\mu_{A\cup B}$	$\mu_{A\cap B}$	μ_{A^c}	$\mu_{A\cap A^c}$	μ_{B^c}	$\mu_{B\cap B^c}$
1	0,9	0,3	0,9	0,3	0,1	0,1	0,7	0,3
2	0,1	0,8	0,8	0,1	0,9	0,1	0,2	0,2
3	1	0,5	1	0,5	0	0	0,5	0,5
4	0,7	0,2	0,7	0,2	0,3	0,3	0,8	0,2
5	0,4	0,9	0,9	0,4	0,6	0,4	0,1	0,1

Na primeira linha, a coluna $\mu_{A\cap A^c}$ igual a 0,1 indica que a atleta 1 está tanto no grupo das fadigadas quanto no das não fadigadas, porém, como $\mu_A(1) > \mu_{A^c}(1)$, isso mostra que a atleta pertence mais ao conjunto das fadigadas do que ao das não fadigadas. Isso mostra que, na teoria fuzzy, dado A subconjunto fuzzy, nem sempre ocorre $A \cap A^c = \varnothing$ como na teoria clássica.

2.3 Normas Triangulares

Nesta seção são apresentadas as definições de normas triangulares de acordo com (JAFELICE; BARROS; BASSANEZI, 2023). Estas normas são generalizações dos operadores união e interseção.

Definição 2.4. *Uma co-norma triangular (s-norma) é uma operação binária denominada* $s : [0,1] \times [0,1] \to [0,1]$*, denotada por* $s(x,y) = xsy$*, satisfazendo as seguintes propriedades:*

(a) Comutatividade: $xsy = ysx$*;*

(b) Associatividade: $xs(ysz) = (xsy)sz$*;*

(c) Monotonicidade: Se $x \leqslant y$ *e* $w \leqslant z$ *então* $xsw \leqslant ysz$*;*

(d) Condições de Fronteira: $xs0 = x$ *e* $xs1 = 1$*.*

Exemplo 2.3. *A seguir, são apresentadas algumas s-normas presentes na literatura.*

1. *União Padrão:* $s : [0,1] \times [0,1] \to [0,1]$*, em que* $xsy = \max(x,y)$*;*
2. *Soma Algébrica:* $s : [0,1] \times [0,1] \to [0,1]$*, em que* $xsy = x + y - xy$*;*
3. *Soma Limitada:* $s : [0,1] \times [0,1] \to [0,1]$*, em que* $xsy = \min(1, x+y)$*;*
4. *União Drástica:* $s : [0,1] \times [0,1] \to [0,1]$*, em que* $xsy = \begin{cases} x, & \text{se } y = 0 \\ y, & \text{se } x = 0 \\ 1, & \text{caso contrário} \end{cases}$.

Definição 2.5. *Uma norma triangular (t-norma) é uma operação binária* $s : [0,1] \times [0,1] \to [0,1]$*, denotada por* $t(x,y) = xty$*, satisfazendo as seguintes propriedades:*

(a) Comutatividade: $xty = ytx$*;*

(b) Associatividade: $xt(ytz) = (xty)tz$*;*

(c) Monotocidade: Se $x \leqslant y$ *e* $w \leqslant z$ *então* $xtw \leqslant ytz$*;*

(d) Condições de Fronteira: $xt0 = 0$ *e* $xt1 = x$*.*

Exemplo 2.4. *A seguir, são apresentadas algumas t-normas presentes na literatura.*

1. *Intersecção Padrão:* $t : [0,1] \times [0,1] \to [0,1]$*, em que* $x\ t\ y = \min(x,y)$*;*
2. *Produto Algébrico:* $t : [0,1] \times [0,1] \to [0,1]$*, em que* $x\ t\ y = xy$*;*
3. *Diferença Limitada:* $t : [0,1] \times [0,1] \to [0,1]$*, em que* $x\ t\ y = \max(0, x+y-1)$*;*
4. *Intersecção Drástica:* $t : [0,1] \times [0,1] \to [0,1]$*, em que* $x\ t\ y = \begin{cases} x, & \text{se } y = 1 \\ y, & \text{se } x = 1 \\ 0, & \text{caso contrário} \end{cases}$.

2.4 Níveis de um Conjunto Fuzzy

O conceito de nível de um conjunto fuzzy permite definir conjuntos tomando como referência valores de pertinência α específicos, sendo de extrema importância na teoria dos conjuntos fuzzy.

Definição 2.6. *Sejam U um conjunto universo, $A \subset U$ um conjunto fuzzy e $\alpha \in (0, 1]$. O α-nível de A em U é definido pelo conjunto clássico*

$$[A]^{\alpha} = \{x \in X | \mu_A(x) \geqslant \alpha\}.$$

Definição 2.7. *Sejam U um conjunto universo e $A \subset U$ um conjunto fuzzy. O suporte de A são todos os elementos de U que possuem grau de pertinência de A diferente de zero, ou seja,*

$$supp(A) = \{x \in U \mid \mu_A(x) > 0\}.$$

Dessa forma, é possível definir o nível zero de um conjunto fuzzy A.

Definição 2.8. *Sejam U um espaço topológico e $A \subset U$ um conjunto fuzzy. O nível zero de A é definido pelo fecho topológico do suporte de A, ou seja,*

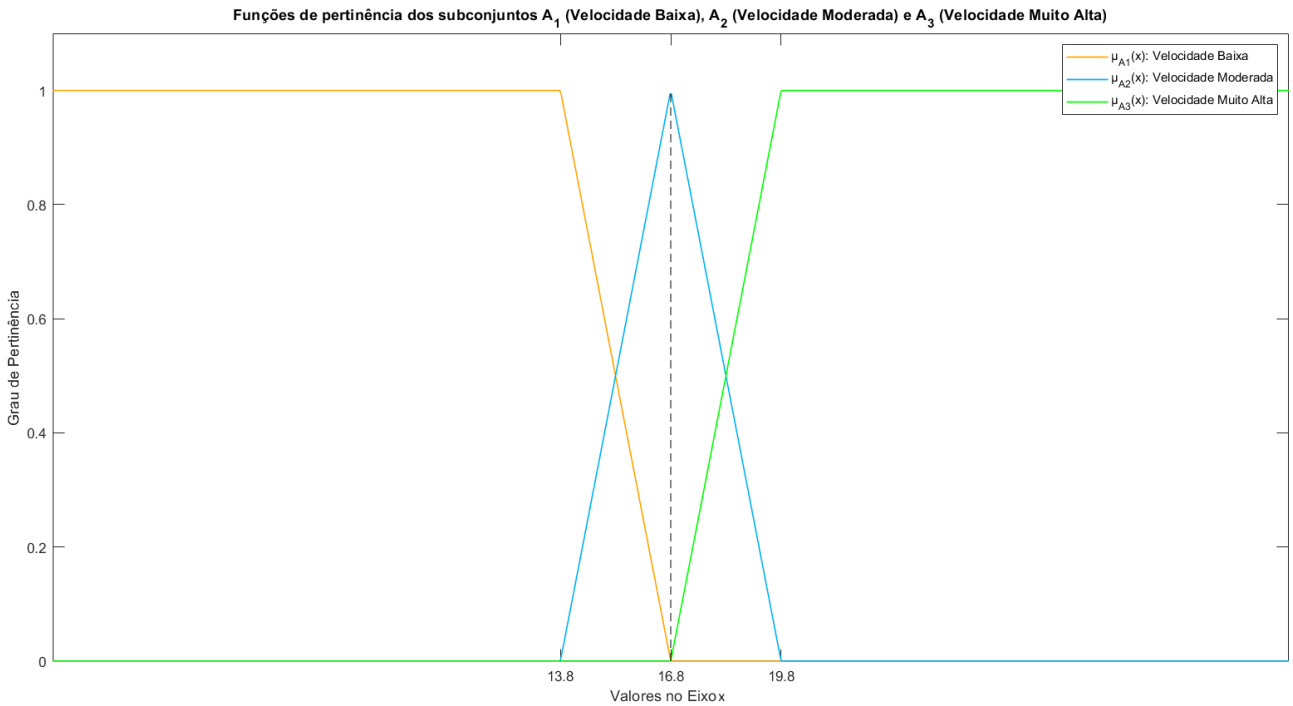
$$[A]^0 = \overline{supp(A)}.$$

O exemplo a seguir ilustra o cálculo do α-nível de um conjunto fuzzy.

Exemplo 2.5. *Seja U o conjunto das velocidades alcançadas por atletas de futebol como no Exemplo 2.1. Seja A_1 o subconjunto das velocidades consideradas "baixas", sendo aquelas entre 0 e 13,8; A_2 o subconjunto das velocidades "moderadas", entre 13,8 e 19,8; e A_3 o subconjunto das velocidades "muito altas", sendo aquelas maiores que 16,8.*

As funções de pertinência dos conjuntos A_1, A_2 e A_3 são modeladas por funções do tipo trapezoidal, triangular e trapezoidal, respectivamente, como mostrado na Figura 2.1.

Figura 2.1: Funções de pertinência dos subconjuntos A_1 (Velocidade Baixa), A_2 (Velocidade Moderada) e A_3 (Velocidade Muito Alta).



Fonte: Autoria própria.

Tome o conjunto A_2 das velocidades "moderadas" como referência. Sua função de pertinência é do tipo triangular, sua expressão é dada por,

$$\mu_{A_2}(x) = \begin{cases} 0, & se\ 0 \leqslant x < 13{,}8 \\ \dfrac{x - 13{,}8}{3}, & se\ 13{,}8 \leqslant x < 16{,}8 \\ \dfrac{19{,}8 - x}{3}, & se\ 16{,}8 \leqslant x < 19{,}8 \\ 0, & se\ x \geqslant 19{,}8 \end{cases}.$$

Pela Definição 2.6, $[A_2]^\alpha = \{x \in X | \mu_{A_2}(x) \geqslant \alpha\}$, assim para $\alpha = 1$, tem-se $[A_2]^1 = 16{,}8$. Para encontrar $[A_2]^\alpha$, para qualquer $\alpha \in (0{,}1]$, é utilizada a expressão da função de pertinência de A_2. A fim de encontrar $x \in X$ tal que $\mu_{A_2}(x) \geqslant \alpha$:

- *Para $13{,}8 \leqslant x < 16{,}8$,*

$$\frac{x - 13{,}8}{3} \geqslant \alpha$$
$$x \geqslant 13{,}8 + 3\alpha \tag{2.2}$$

- *Para $16{,}8 \leqslant x < 19{,}8$,*

$$\frac{19{,}8 - x}{3} \geqslant \alpha$$
$$x \leqslant 19{,}8 - 3\alpha \tag{2.3}$$

Dessa forma, pelas equações (2.2) e (2.3), tem-se

$$[A_2]^\alpha = \{x \in X | \mu_{A_2}(x) \geqslant \alpha\} = [13{,}8 + 3\alpha\ ,\ 19{,}8 - 3\alpha].$$

Note que $supp(A_2) = (13{,}8\ ,\ 19{,}8)$, logo, $[A_2]^0 = [13{,}8\ ,\ 19{,}8]$.

2.5 Números Fuzzy

Números fuzzy são considerados uma generalização dos números reais clássicos (BEDE, 2013). Sua criação surge da necessidade de se realizar cálculos dentro da teoria fuzzy, assim como na teoria clássica, porém agora com quantidades imprecisas. O conceito de número fuzzy é fundamental em diversas aplicações de conjuntos fuzzy e lógica fuzzy.

Definição 2.9. *Um conjunto fuzzy N é chamado de "número fuzzy" quando o conjunto universo U, em que N está definido, é o conjunto dos números reais, ou seja, $U = \mathbb{R}$ e sua função de pertinência $\mu_N : \mathbb{R} \to [0, 1]$ satisfaz as seguintes propriedades:*

(a) $\mu_N(x) = 1$ para pelo menos um valor x do $supp(N)$;

(b) $[N]^\alpha$ é um intervalo fechado para todo $\alpha \in (0, 1]$;

(c) O suporte de N é limitado.

Exemplo 2.6. *Observe que todo número real r é um caso particular de número fuzzy cuja função de pertinência $\mu_r(x)$ é sua função característica*

$$\mu_r(x) = \begin{cases} 1, & se\ x = r \\ 0, & se\ x \neq r \end{cases}.$$

Outros números fuzzy amplamente utilizados na teoria são os números triangulares, trapezoidais e gaussianos.

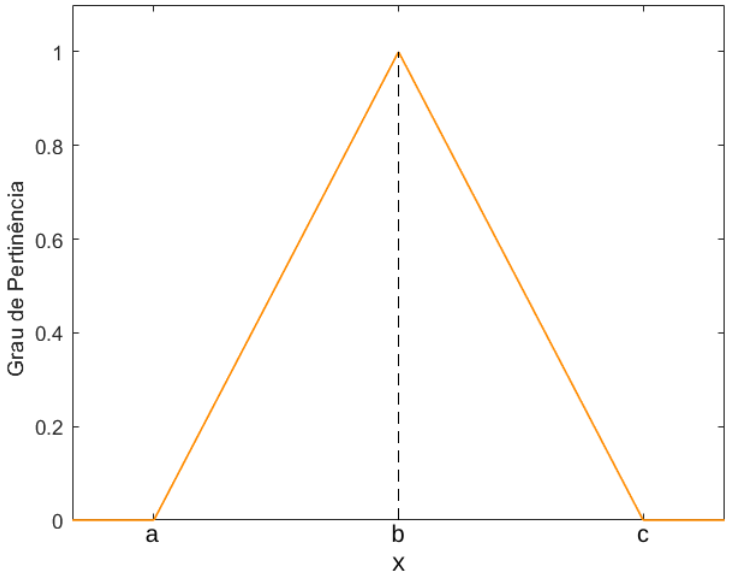
Definição 2.10. *Um número fuzzy A é dito triangular se sua função de pertinência possui a seguinte forma:*

$$\mu_A(x) = \begin{cases} \dfrac{x-a}{b-a}, & \text{se } a \leqslant x < b \\ \dfrac{-x+c}{c-b}, & \text{se } b \leqslant x \leqslant c \\ 0, & \text{caso contrário} \end{cases},$$

para a, b, $c \in \mathbb{R}$.

Observe que o gráfico da função de pertinência de um número fuzzy triangular forma um triângulo com o eixo x, tendo o intervalo $[a, c]$ como base e o ponto $(b, 1)$ como vértice. Um conjunto fuzzy triangular pode ser denotado como $A = (a; b; c)$, sendo a, b e c as abscissas dos vértices do triângulo, ver Figura 2.2.

Figura 2.2: Número fuzzy triangular.



Fonte: Autoria própria.

Definição 2.11. *Um número fuzzy A é dito trapezoidal se sua função de pertinência possui a seguinte forma:*

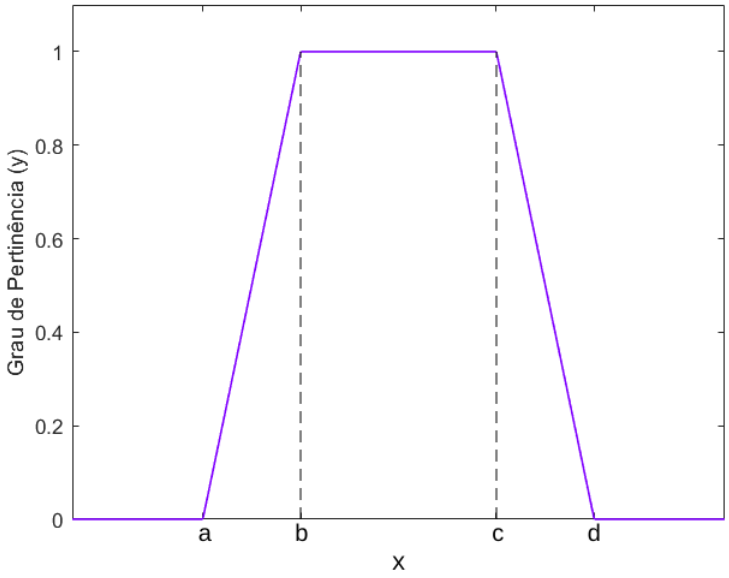
$$\mu_A(x) = \begin{cases} \dfrac{x-a}{b-a}, & \text{se } a \leqslant x < b \\ 1, & \text{se } b \leqslant x \leqslant c \\ \dfrac{d-x}{d-c}, & \text{se } c < x \leqslant d \\ 0, & \text{caso contrário} \end{cases},$$

para a, b, $c \in \mathbb{R}$.

Observe que o gráfico da função de pertinência de um número fuzzy trapezoidal forma um trapézio com o eixo x, possuindo o intervalo $[a, d]$ como base maior e o intervalo $[b, c]$ na altura

1 como base menor. Um conjunto fuzzy trapezoidal pode ser denotado como $A = (a; b; c; d)$, ver Figura 2.3.

Figura 2.3: Número fuzzy trapezoidal.



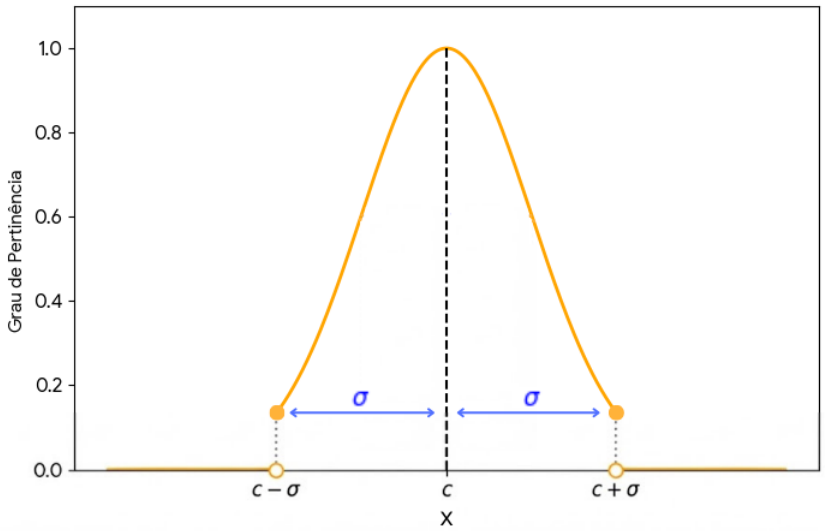
Fonte: Autoria própria.

Definição 2.12. *Um número fuzzy A é dito gaussiano se sua função de pertinência possui a seguinte forma:*

$$\mu_A(x) = \begin{cases} e^{-\frac{(x-c)^2}{2\sigma^2}}, & se\ c - \sigma \leqslant x \leqslant c + \sigma\ , \\ 0, & caso\ contrário \end{cases}$$

sendo $c \in \mathbb{R}$ o centro e $\sigma \in \mathbb{R}$ a largura da função.

Figura 2.4: Número fuzzy gaussiano.



Fonte: Autoria própria.

Observe que o gráfico da função de pertinência de um número fuzzy gaussiano possui o formato de um sino, com o ponto $(c,1)$ sendo o seu pico. Um conjunto fuzzy gaussiano pode ser denotado por $A = (c;\sigma)$, sendo c o centro da curva e σ a largura, ver Figura 2.4.

Na seção a seguir, são apresentados os componentes do SBRF, um diagrama deste sistema e o método de inferência do tipo Takagi-Sugeno.

2.6 Sistema Baseado em Regras Fuzzy

Um Sistema Baseado em Regras Fuzzy (SBRF) é um sistema que utiliza a lógica fuzzy para tomar decisões e resolver problemas, sendo utilizado em processos de modelagem, controle e classificação. O SBRF possui como entradas conjuntos fuzzy e sua saída passa por um processo de defuzzificação para se tornar um valor real. Tais sistemas foram desenvolvidos com o intuito de imitar o comportamento humano (JAFELICE; BARROS; BASSANEZI, 2023).

O sistema opera com um conjunto de regras fuzzy, as quais usualmente possuem a forma (JAFELICE; BARROS; BASSANEZI, 2023):

Se (um conjunto de condições é satisfeito) então (um conjunto de consequências pode ser inferido).

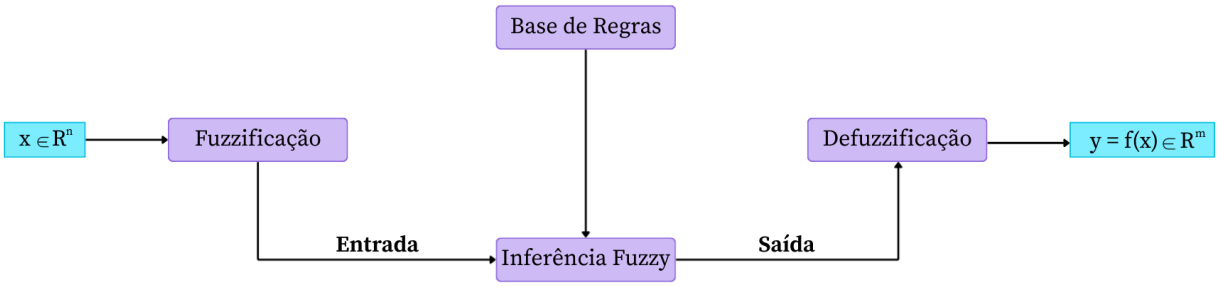
Por exemplo:

Se (a fadiga informada pela atleta é ELEVADA) então (o risco de lesão da atleta é ALTO).

Dado um conjunto de dados, sendo $x \in \mathbb{R}^n$ o vetor de entrada em um espaço n-dimensional e $y \in \mathbb{R}^m$ o vetor de saída em um espaço m-dimensional. Dessa forma, um SBRF pode ser interpretado como um mapeamento entre a entrada e a saída do sistema $y = f(x)$ (JAFELICE; BARROS; BASSANEZI, 2023).

A estrutura do SBRF é composta por 4 componentes: a fuzzificação dos dados de entrada; a base de regras, em que estão localizadas as regras fuzzy do sistema; um método de inferência fuzzy, em que as regras fuzzy são traduzidas para uma linguagem matemática; e a defuzzificação, em que é calculada a saída real do sistema (JAFELICE; BARROS; BASSANEZI, 2023). Na Figura 2.5 é ilustrado como os componentes mencionados anteriormente estão interligados.

Figura 2.5: Diagrama de um SBRF.



Fonte: Autoria própria.

1- Fuzzificação

Nesta componente as entradas do sistema são transformadas em conjuntos fuzzy, processo conhecido por fuzzificação. Nesta etapa, a visão de um especialista da área de estudo em questão é de extrema importância, auxiliando nos processos de construção das funções de pertinência do sistema.

2- Base de Regras

É considerado o núcleo do SBRF, juntamente com a máquina de inferência. Nesta componente estão localizadas a coleção de regras fuzzy na forma "Se (condição), então (consequência)". Cada regra possui a seguinte forma:

$$\text{Se } x_1 \text{ é } A_1 \text{ e } \cdots \text{ e } x_n \text{ é } A_n \text{ então } y_1 \text{ é } B_1 \text{ e } \cdots \text{ e } y_m \text{ é } B_m,$$

sendo $A_1, \cdots, A_n$ e $B_1, \cdots, B_m$ conjuntos fuzzy relacionados as variáveis linguísticas de entrada e saída respectivamente. As variáveis linguísticas que formam as condições e consequências, como por exemplo "PEQUENO", "MÉDIO" ou "GRANDE", podem ser descritas pelo especialista da área.

3- Método de Inferência Fuzzy

Nesta componente, cada regra fuzzy é escrita matematicamente utilizando técnicas da teoria fuzzy. Assim, são selecionados os operadores matemáticos, como as t-normas, as s-normas e as regras de inferências para definir a relação fuzzy que modela a base de regras. Nesta etapa é fornecida a saída a partir de cada entrada fuzzy e da relação presente na base de regras. Neste trabalho é apresentado o método de inferência fuzzy do tipo Takagi-Sugeno, que é utilizado no DENFIS e no SBC/FCM, detalhados na Seção 2.7.

3.1 - Método de Takagi-Sugeno

Seja uma base de regras com m regras e q variáveis. O Método de Takagi-Sugeno (TAKAGI; SUGENO, 1985) possui regras da forma:

$$\text{SE } x_1 \text{ é } R_{11} \text{ e } \cdots \text{ e } x_q \text{ é } R_{1q} \text{ ENTÃO } y_1 = f_1(x_1, \cdots, x_q)$$

$$\text{SE } x_1 \text{ é } R_{21} \text{ e } \cdots \text{ e } x_q \text{ é } R_{2q} \text{ ENTÃO } y_2 = f_2(x_1, \cdots, x_q)$$

$$\vdots$$

$$\text{SE } x_1 \text{ é } R_{m1} \text{ e } \cdots \text{ e } x_q \text{ é } R_{mq} \text{ ENTÃO } y_m = f_m(x_1, \cdots, x_q),$$

em que o consequente da i-ésima regra y_i é uma função das variáveis de entrada $x_1, \cdots, x_q$. Como, por exemplo, uma função linear, $y_i = f_i(x_1, \cdots, x_q) = a_1 x_1 + \cdots + a_q x_q + c$, para $a_1, \cdots, a_q, c \in \mathbb{R}$.

O método de Takagi-Sugeno utiliza em cada regra o operador "E", podendo este ser modelado pelos operadores mínimo ou produto.

Na Figura 2.6 é ilustrado como uma saída y de um sistema do método de Takagi-Sugeno é gerada a partir das entradas reais x_1 e x_2 que ativam as regras fuzzy 1 e 2:

$$\text{Regra 1: SE } x_1 \text{ é } A_1 \text{ e } x_2 \text{ é } B_1 \text{ ENTÃO } y_1 = a_1^1 \cdot x_1 + a_2^1 \cdot x_2 + c^1;$$

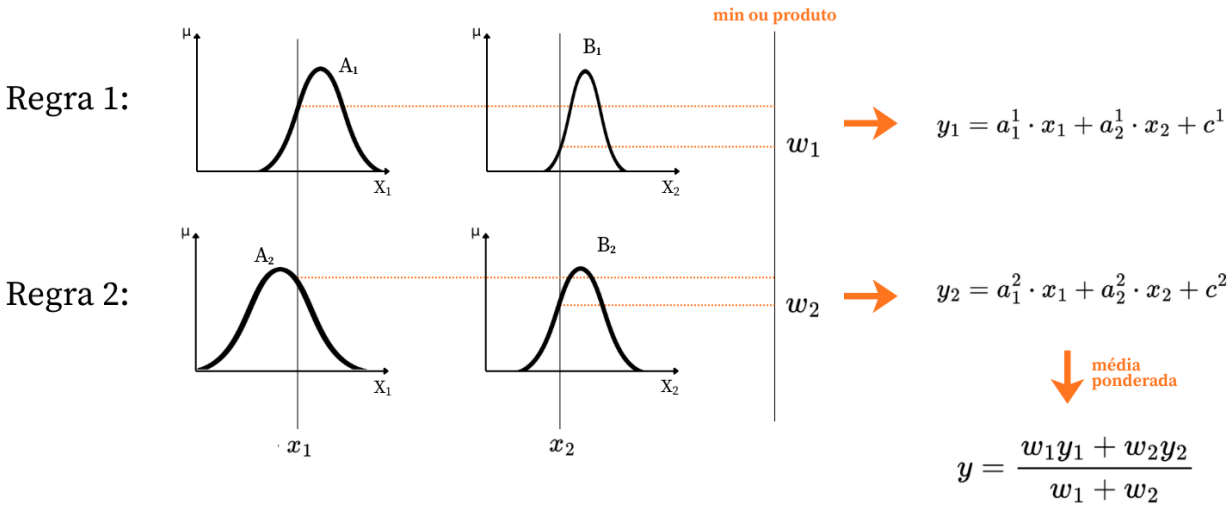
$$\text{Regra 2: SE } x_1 \text{ é } A_2 \text{ e } x_2 \text{ é } B_2 \text{ ENTÃO } y_2 = a_1^2 \cdot x_1 + a_2^2 \cdot x_2 + c^2.$$

4- Defuzzificação

No geral, a saída gerada pelo método de inferência de um sistema fuzzy é um conjunto fuzzy, dessa forma, o processo de defuzzificação surge para transformar esse conjunto fuzzy em uma saída real. Assim, é necessário a escolha de um método de defuzzificação para realizar a transformação do conjunto fuzzy em um número real. A média ponderada é utilizada no método de Takagi-Sugeno, o processo de defuzzificação é ilustrado na Figura 2.6.

Na seção a seguir, são apresentados os sistemas de inferência neuro-fuzzy utilizados neste trabalho. A seção apresenta a teoria e um exemplo para a compreensão de ambos os métodos, o DENFIS e o SBC/FCM.

Figura 2.6: Processo do método de inferência fuzzy do tipo Takagi-Sugeno.



Fonte: Autoria própria.

2.7 Sistemas de Inferência Neuro-Fuzzy via Clusterização

Cluster, ou agrupamento, é a tarefa de identificar características similares em um conjunto de dados e separá-los em *clusters*, grupos com padrões semelhantes (GÉRON, 2017). Para isso, é necessário concretizar o que significa que duas ou mais observações possuam características semelhantes ou diferentes (JAMES *et al.*, 2021). Essa consideração é feita com base no conjunto de dados disponíveis.

Não há uma definição universal sobre a organização dos *clusters*, tais agrupamentos são formados de acordo com o contexto do problema em questão. Diferentes algoritmos formam *clusters* de maneiras diferentes. Alguns procuram características centralizadas em torno de um ponto específico, conhecido como centróide. Outros são hierárquicos, procurando *clusters* de *clusters* (GÉRON, 2017).

Nesta seção, são apresentados dois métodos de inferência que formam suas regras fuzzy a partir de *clusters*, o DENFIS e o Método de Agrupamento Subtrativo com Fuzzy C-Means.

2.7.1 DENFIS

Desenvolvida no início dos anos 2000 pelo cientista búlgaro Nikola Kasabov, a abordagem DENFIS, abreviação de *Dynamic Evolving Neural Fuzzy Inference System* (Sistema de Inferência Neuro-Fuzzy Dinâmico Evolutivo) (KASABOV; SONG, 2002), é considerada um dos desenvolvimentos mais importantes na área dos sistemas neuro-fuzzy (LUGHOFER, 2011).

O sistema foi originalmente criado a partir da abordagem EFuNN (*Evolving Fuzzy Neural Network*), a qual foi motivada pela arquitetura ECOS (*Evolving Connectionist Systems*), traduzida como Sistemas Conexionistas Evolutivos. ECOS são abordagens no sistema de modelagem dinâmica não-linear, as quais foram inspiradas no comportamento neural biológico e na evolução dos humanos ao longo de sua vida (MENDES *et al.*, 2012). Um sistema conexionista evolutivo evolui e se adapta ao longo do tempo, aprendendo de forma contínua a partir da sua interação com o ambiente. A estrutura e os parâmetros de um ECOS se ajustam conforme novos dados são inseridos no modelo (KASABOV, 1999).

Dada uma rede neural tradicional, ou seja, um modelo estático, após seu treinamento, esta não se modifica e pode ser utilizada para prever novos dados. No caso de uma nova

observação com características diferentes do conjunto de treinamento, esse modelo pode não realizar uma boa previsão para essa nova observação. Enquanto, um modelo que utiliza a abordagem evolutiva se adapta a essa nova informação, a utilizando para renovar seus parâmetros e sua estrutura. O sistema DENFIS utiliza a abordagem evolutiva do ECOS para ajustar sua estrutura dinamicamente.

Pelo seu processo de clusterização, o DENFIS é muito utilizado em modelagens dinâmicas, como preferências de clientes e algoritmos de recomendação. Sabendo que preferências de clientes podem mudar rapidamente, é necessária uma predição em tempo real durante o lançamento de um produto (JIANG *et al.*, 2019). No contexto da predição de lesão, o mesmo pode ocorrer. Uma vez que a predição é realizada com variáveis de bem-estar e carga de treinamento, a cada treino realizado pela atleta, seus dados de entrada são atualizados e, com isso, o seu risco de lesão pode mudar.

Outra diferença do sistema DENFIS em relação às redes neurais tradicionais e outras abordagens é a forma como a DENFIS prevê uma saída dada uma nova observação (LUGHOFER, 2011). Este utiliza o "aprendizado preguiçoso por amostra", também conhecido como Aprendizagem Baseada em Instâncias (*Instance-Based Learning*). Essa abordagem considera amostras locais próximas ao ponto para construir um pequeno modelo local sob demanda (LUGHOFER, 2011), ou seja, esses modelos adiam seu trabalho de generalização; a construção de um modelo local somente é iniciada no momento em que uma nova observação é fornecida ao sistema. No caso da DENFIS, um modelo local significa uma nova regra fuzzy, desenvolvida a partir de um *cluster*. Dada uma nova amostra para predição, o DENFIS calcula sua distância em relação aos centros dos *clusters* já existentes e seleciona os m *clusters* mais próximos. A partir das m regras formadas por cada *cluster*, é construído um sistema de inferência fuzzy local. O sistema é então utilizado para prever a saída dessa nova amostra, combinando os resultados das m regras selecionadas inicialmente (LUGHOFER, 2011).

Segundo Lughofer (2011), as 4 principais diferenças do sistema DENFIS em relação a outros sistemas neuro-fuzzy são:

1- Utilização do método de inferência fuzzy do tipo Takagi-Sugeno;

2- Aprendizagem local dos parâmetros;

3- Utilização de funções de pertinência triangular;

4- Evolução de regras (neurônios) utilizando uma abordagem baseada em agrupamento: evoluir um novo agrupamento significa evoluir uma nova regra.

A evolução das regras por agrupamento é possível utilizando o ECM (*Evolving Clustering Method*), também conhecido como Método de Clusterização/Agrupamento Evolutivo.

Método de Clusterização Evolutiva (ECM)

O ECM (*Evolving Clustering Method*), traduzido como método de clusterização evolutiva, possui uma função primordial em processos de aprendizagem de modelos evolutivos neuro-fuzzy (MENDES *et al.*, 2012), sendo um método de agrupamento baseado na distância máxima. Por ser uma técnica caracterizada como on-line, o ECM não utiliza todos os dados de treinamento de uma única vez. Seu mecanismo de funcionamento busca particionar, de forma iterativa, os dados de entrada em subconjuntos, denominados *clusters*, com o objetivo de criar regras de inferência fuzzy (KASABOV; SONG, 2002).

Esse método possui dois modos: o método de agrupamento evolutivo on-line (ECM), utilizado na versão on-line do DENFIS, e o método de agrupamento evolutivo off-line com minimização restrita (ECMc), sendo este último uma extensão da versão on-line (KASABOV;

SONG, 2002). O ECM consegue uma boa performance como um método de agrupamento on-line, mas no pacote utilizado para rodar o sistema DENFIS neste trabalho (pacote `frbs` no *software* R), o método é utilizado em seu modo off-line.

Método de Agrupamento Evolutivo On-line

Nesse processo de agrupamento, o modelo é iniciado sem nenhum *cluster*. Após o processamento da primeira observação do conjunto de treinamento, esse ponto se torna o centro c_1 do primeiro *cluster* C_1, possuindo um raio inicial r_1 igual a 0. Conforme mais exemplos são processados pelo modelo, um por um, alguns *clusters* são criados e outros são atualizados, juntamente com a posição de seus centros e tamanhos de raios (KASABOV; SONG, 2002).

Nesse método, um valor limite de distância *Dthr* entre 0 e 1 é pré-definido pelo usuário. Esse parâmetro é utilizado para a tomada de decisão do modelo sobre a criação de novos *clusters* ou a atualização de *clusters* existentes.

Geometricamente, o parâmetro *Dthr* pode ser interpretado como o maior raio que um *cluster* pode assumir, neste contexto, utilizando a distância euclidiana, os *clusters* podem ser interpretados como hiperesferas de diâmetro máximo igual a $2 \cdot Dthr$, formadas no espaço de entrada q-dimensional, sendo q a quantidade de variáveis de entrada (MENDES *et al.*, 2012).

Definição 2.13 (Distância Euclidiana)**.** *Dado dois vetores em um espaço de dimensão q,* $x = (x_1, \cdots, x_q)$ *e* $y = (y_1, \cdots, y_q)$*, a distância euclidiana entre* x *e* y *é definida por,*

$$d(x,y) = \left(\sum_{i=1}^{q} (x_i - y_i)^2 \right)^{\frac{1}{2}}. \tag{2.4}$$

O algoritmo completo para um conjunto de treinamento com n observações do processo ECM on-line é detalhado a seguir, com base nas referências (KASABOV; SONG, 2002) e (MENDES *et al.*, 2012):

Passo 1: Processar a primeira amostra do conjunto de dados e criar o primeiro *cluster* C_1, atribuindo as coordenadas do seu centro c_1 às coordenadas da primeira amostra do conjunto de treinamento x_1, ou seja, $c_1 = x_1$ e determine seu raio r_1 como 0, $r_1 = 0$.

Passo 2: Processar a próxima amostra de entrada x_k, para $k \in 2, \cdots, n$, se disponível; caso contrário, pare.

Passo 3: Calcular a distância entre x_k e todos os m centros de *cluster* já criados $d(x_k, c_j)$, para $j = 1, \cdots, m$, sendo d a distância euclidiana.

Passo 4: Guardar a menor das distâncias calculadas no Passo 3.

$$D = min\{d(x_k, c_j)\}, \text{ para } j = 1, \cdots, m.$$

e calcular $S_k = D + r_j$, sendo r_j o raio do cluster mais próximo ao centro c_j.

Passo 5: Se $D < r_j$, então a amostra atual x_k pertence ao cluster C_j; neste caso, nenhum novo cluster é criado nem nenhum cluster existente é atualizado. Retorne ao Passo 2. Caso contrário, passar para o próximo passo.

Passo 6: (Criação de um novo *cluster*) Se $S_k > 2 \cdot D_{thr}$, então x_k não pertence a nenhum *cluster* existente. Um novo *cluster* é criado, conforme descrito no Passo 1; retorne ao Passo 2.

Passo 7: (Atualização de um *cluster*) Se $S_k < 2 \cdot D_{thr}$, então o *cluster* C_j é expandido e seu centro é movido em direção a x^k. O novo raio atualizado r_j^{novo} é definido como $\frac{S_k}{2}$ e o novo centro c_j^{novo} é localizado no ponto da reta que conecta x_k a c_j de modo que a distância do novo centro c_j^{novo} ao ponto x_k seja aproximadamente igual ao novo raio r_j^{novo}. Retorne ao Passo 2.

Para ilustrar a criação de *clusters* utilizando o ECM on-line, considere o seguinte conjunto de dados da Tabela 2.2, o qual contém as variáveis altura (cm) e peso (kg) de 7 atletas.

Tabela 2.2: Altura x Peso de atletas para exemplo de ECM on-line.

Atleta	Altura (cm)	Peso (kg)
x_1	177	64
x_2	179	71
x_3	174	66
x_4	178	65
x_5	174	68
x_6	180	70
x_7	175	67

Fonte: Autoria própria.

Apresentam-se, a seguir, os cálculos dos *clusters* para os dados da Tabela 2.2, obtidos utilizando o algoritmo do ECM on-line de maneira iterativa. Neste exemplo, foi utilizado o limiar de distância $Dthr = 1$.

Iteração 1:

Passo 1 Processamento da primeira amostra $x_1 = (177, 64)$: Nesta etapa, é criado o primeiro *cluster* C_1 de raio 0 e centro $c_1 = x_1 = (177, 64)$;

Passo 2 Processamento da próxima amostra: $x_2 = (179, 71)$;

Passo 3 Cálculo das distâncias de x_2 aos centros dos *clusters* existentes:

$$d(x_2, c_1) = \sqrt{(179 - 177)^2 + (71 - 64)^2} = 7{,}28;$$

Passo 4 Cálculo da menor distância D e S_2:

$$D = min\{d(x_2, c_1)\} = 7{,}28;$$

Como a menor das distâncias é em relação ao *cluster* 1, então

$$S_2 = D + r_1 = 7{,}28 + 0 = 7{,}28;$$

Passo 5 Avaliando o pertencimento de x_2 ao *cluster* C_1: Como $D < r_1$ ($7{,}28 < 0$) é falso, x_2 não pertence diretamente ao *cluster* C_1;

Passo 6 Avaliando a criação de um novo *cluster*: Como $S_2 > 2 \cdot Dthr$ ($7{,}28 > 2 \cdot 1$) é verdadeiro, então x_2 não pertence a nenhum *cluster* existente, e um novo agrupamento C_2 é criado conforme descrito no Passo 1, com raio 0 e centro $c_2 = x_2 = (179, 71)$. Retorno ao Passo 2.

Iteração 2:

Passo 2 Processamento da próxima amostra: $x_3 = (174, 66)$;

Passo 3 Cálculo das distâncias de x_3 aos centros dos *clusters* existentes:

$$d(x_3, c_1) = 3{,}605;$$

$$d(x_3, c_2) = 7{,}070;$$

Passo 4 Cálculo da menor distância D e S_3:

$$D = min\{d(x_3, c_1), d(x_3, c_2)\} = 3{,}605;$$

Como a menor das distâncias é em relação ao *cluster* 1, então

$$S_3 = D + r_1 = 3{,}605 + 0 = 3{,}605;$$

Passo 5 Avaliando o pertencimento de x_3 ao *cluster* C_1: Como $D < r_1$ $(3{,}605 < 0)$ é falso, x_3 não pertence diretamente ao *cluster* C_1;

Passo 6 Avaliando a criação de um novo *cluster*: Como $S_3 > 2 \cdot Dthr$ $(3{,}605 > 2 \cdot 1)$ é verdadeiro, então é criado um novo agrupamento conforme descrito no Passo 1, com raio 0 e centro $c_3 = x_3 = (174, 66)$. Retorno ao Passo 2.

Iteração 3:

Passo 2 Processamento da próxima amostra: $x_4 = (178, 65)$;

Passo 3 Cálculo das distâncias de x_4 aos centros dos *clusters* existentes:

$$d(x_4, c_1) = 1{,}414;$$

$$d(x_4, c_2) = 6{,}082;$$

$$d(x_4, c_3) = 4{,}123;$$

Passo 4 Cálculo da menor distância D e S_4:

$$D = min\{d(x_4, c_1), d(x_4, c_2), d(x_4, c_3)\} = 1{,}414;$$

Como a menor das distâncias é em relação ao *cluster* 1, então

$$S_4 = D + r_1 = 1{,}414 + 0 = 1{,}4145;$$

Passo 5 Avaliando o pertencimento de x_4 ao *cluster* C_1: Como $D < r_1$ $(1{,}414 < 0)$ é falso, x_4 não pertence diretamente ao *cluster* C_1;

Passo 6 Avaliando a criação de um novo *cluster*: Como $S_4 > 2 \cdot Dthr$ $(1{,}414 > 2 \cdot 1)$ é falso, não é criado um novo agrupamento para x_4;

Passo 7 Avaliando a atualização de um novo *cluster* existente: Como $S_4 < 2 \cdot Dthr$ $(1{,}414 < 2 \cdot 1)$ é verdadeiro, então o *cluster* C_1 é atualizado, expandindo seu raio para $r_1 = \frac{S_4}{2} = 0{,}707$ e seu novo centro c_1^{novo} é escolhido de modo que pertença a reta que conecta x_4 ao antigo centro c_1 e a distância do novo centro ao ponto x_4 seja aproximadamente igual ao novo raio r_1. Assim, $c_1^{novo} = (177{,}5\ ,\ 64{,}5)$. Retorno ao Passo 2.

O passo a passo do algoritmo continua até que todas as observações do conjunto de dados sejam processadas. Dessa forma, o resultado da aplicação do ECM nos dados da Tabela 2.2 são os *clusters* obtidos.

Tabela 2.3: *Clusters* resultantes da aplicação do algoritmo ECM aos dados da Tabela 2.2.

Cluster	Centro	Raio	Atletas	Descrição
C_1	(177,5 , 64,5)	0,707	x_1, x_4	Atletas de altura média e leves
C_2	(179,5 , 70,5)	0,707	x_2, x_6	Atletas altas e pesadas
C_3	$(174, 67)$	1	x_3, x_5, x_7	Atletas baixas e de peso médio

Método de Agrupamento Evolutivo com Restrições (ECM off-line)

O método de agrupamento evolutivo com restrições, do inglês *Evolving Clustering Method with Constraints* (ECMc), também conhecido como a versão off-line do ECM, realiza um processo de otimização nos centros dos *clusters* gerados pelo ECM (KASABOV; SONG, 2002). O intuito do método off-line é particionar um conjunto de p vetores x_i, $i = 1, \cdots, p$ em n *clusters* C_j, $j = 1, \cdots, n$, com o objetivo de encontrar um centro de *cluster* em cada agrupamento, minimizando uma função objetivo com restrições.

Utilizando a distância euclidiana como medida de distância entre um vetor x_i no *cluster* C_j e seu centro c_j, a função objetivo é definida por:

$$J = \sum_{j=1}^{n} J_j = \sum_{j=1}^{n} \left(\sum_{x_i \in C_j} d(x_i, c_j) \right), \tag{2.5}$$

sendo $J_i = \sum_{\mathbf{x_i} \in C_j} d(x_i, c_j)$ a função objetivo dentro do *cluster* C_j, $i = 1, \cdots, p$ e $j = 1, \cdots, n$. As restrições da otimização são definidas da seguinte maneira:

$$d(x_i, c_j) \leqslant D_{thr}, \quad j = 1, 2, \ldots, n. \tag{2.6}$$

Após a partição, os *clusters* são definidos por uma matriz de pertinência binária U, de p linhas e n colunas, onde o elemento u_{ij} é igual a 1 se x_i pertence ao agrupamento C_j e 0 caso contrário. O ponto x_i pertence ao *cluster* C_j apenas se a distância ao centro desse *cluster* for a menor entre todas as distâncias aos outros *clusters* C_k, com $k \neq j$.

Dessa forma, uma vez fixados os centros dos clusters C_j, a minimização de u_{ij} para as Equações (2.5) e (2.6) é obtida da seguinte forma:

$$u_{ij=} \begin{cases} 1, & \text{se } d(x_i, c_j) \leqslant d(x_i, c_k), \text{ para cada } j \neq k \\ 0, & \text{caso contrário.} \end{cases} \tag{2.7}$$

O algoritmo que representa o método do ECMc off-line para determinar os centros dos *clusters* C_j e a matriz de pertinência U, é apresentado por (KASABOV; SONG, 2002) da seguinte maneira:

Passo 1 Inicializar os centros dos *clusters* C_j, $j = 1, \cdots, n$ resultantes do processo de agrupamento ECM on-line;

Passo 2 Determinar a matriz de pertinência U usando a Equação (2.7);

Passo 3 Aplicar o método de minimização restrita (referência (THE MATH WORKS INC., 1996)) com as Equações (2.5) e (2.6) para obter os novos centros de *cluster*;

Passo 4 Calcular a função objetivo J de acordo com a restrição (Equação (2.6)). O algoritmo deve ser interrompido se:

4.1 Se o resultado for menor que um valor de tolerância;

4.2 Se a melhoria em relação à iteração anterior for menor que um certo limiar;

4.3 Se o número de iterações de minimização ultrapassar um certo valor;

Caso contrário, retorne ao Passo 2.

A partir da aplicação do ECM são gerados os *clusters* de um conjunto de dados. No DENFIS, evoluir um novo *cluster* significa evoluir uma nova regra. O centro e o raio de cada agrupamento são utilizados para montar a parte antecedente de uma regra fuzzy. A seguir, é detalhada a formação da base de regras do DENFIS.

Antecedentes das Regras Fuzzy e Funções de Pertinência

O sistema neuro-fuzzy evolutivo DENFIS, tanto em seu modo on-line quanto off-line, utiliza o método de inferência fuzzy do tipo Takagi-Sugeno (KASABOV; SONG, 2002). O método de inferência é composto por m regras fuzzy, representadas da seguinte forma:

$$\text{SE } x_1 \text{ é } A_{11} \text{ e } \cdots \text{ e } x_q \text{ é } A_{1q} \text{ ENTÃO } y_1 = f_1(x_1, \cdots, x_q)$$

$$\text{SE } x_1 \text{ é } A_{21} \text{ e } \cdots \text{ e } x_q \text{ é } A_{2q} \text{ ENTÃO } y_2 = f_2(x_1, \cdots, x_q)$$

$$\cdots$$

$$\text{SE } x_1 \text{ é } A_{m1} \text{ e } \cdots \text{ e } x_q \text{ é } A_{mq} \text{ ENTÃO } y_m = f_m(x_1, \cdots, x_q),$$

sendo q o número de variáveis do conjunto de dados, x_j, $j = 1, \cdots, q$ as variáveis antecedentes definidas sobre o universo X_j, A_{ij} os conjuntos fuzzy para $i = 1, \cdots, m$ e $j = 1, \cdots, q$, definidos por suas funções de pertinência fuzzy

$$\mu_{A_{ij}} : X_j \rightarrow [0,1].$$

Na parte consequente do método de inferência Takagi-Sugeno, tem-se y sendo a variável de saída e f_i funções polinomiais para $i = 1, \cdots, m$.

Tanto para a versão on-line do sistema DENFIS quanto para a off-line são consideradas funções de pertinência triangulares que possuem a seguinte expressão

$$\mu(x_i) = \begin{cases} 0, & \text{se } x_i < a \\ \dfrac{x_i - a}{b - a}, & \text{se } a \leqslant x_i < b \\ \dfrac{c - x_i}{c - b}, & \text{se } b \leqslant x_i \leqslant c \\ 0, & \text{se } x_i > c \end{cases}, \tag{2.8}$$

sendo b o centro do *cluster* c_j, $a = b - d \cdot D_{thr}$, $c = b + d \cdot D_{thr}$, d um parâmetro entre 1.2 e 2 e D_{thr} o limiar de distância do método ECM (KASABOV; SONG, 2002).

Uma vez que a aprendizagem local dos parâmetros das funções de pertinência é realizada utilizando o método ECM, é possível realizar a seguinte correspondência entre os parâmetros a, b e c e os valores gerados pelo algoritmo ECM (MENDES *et al.*, 2012)

$$(a, b, c) = (c_j - r_j, c_j, c_j + r_j),$$

sendo c_j o centro do *cluster* e r_j o raio do *cluster* C_J, quando varia-se adequadamente o hiper-parâmetro d. A correspondência é desenvolvida uma vez que nas funções de pertinência triangulares o ponto b é onde se tem a pertinência total 1, e em um *cluster* o ponto mais próximo do centro c_j é o próprio c_j. Para os parâmetros a e c, nas funções triangulares, a base

(a, c) define a região onde a pertinência é maior que 0, podendo ser relacionada à fronteira do *cluster*.

Para ilustrar a criação do antecedente das regras fuzzy e das funções de pertinência, considere os *clusters* da Tabela 2.3 criados pela aplicação do ECM no conjunto de dados da Tabela 2.2. O *cluster* C_1, de centro $c_1 = (177{,}5\ ,\ 64{,}5)$ e raio $r = 0{,}707$, gera a seguinte regra fuzzy 1,

$$\text{SE altura é } A_{11} \text{ E peso é } A_{12} \text{ ENTÃO } y_1 = f_1(x_1, \cdots, x_q),$$

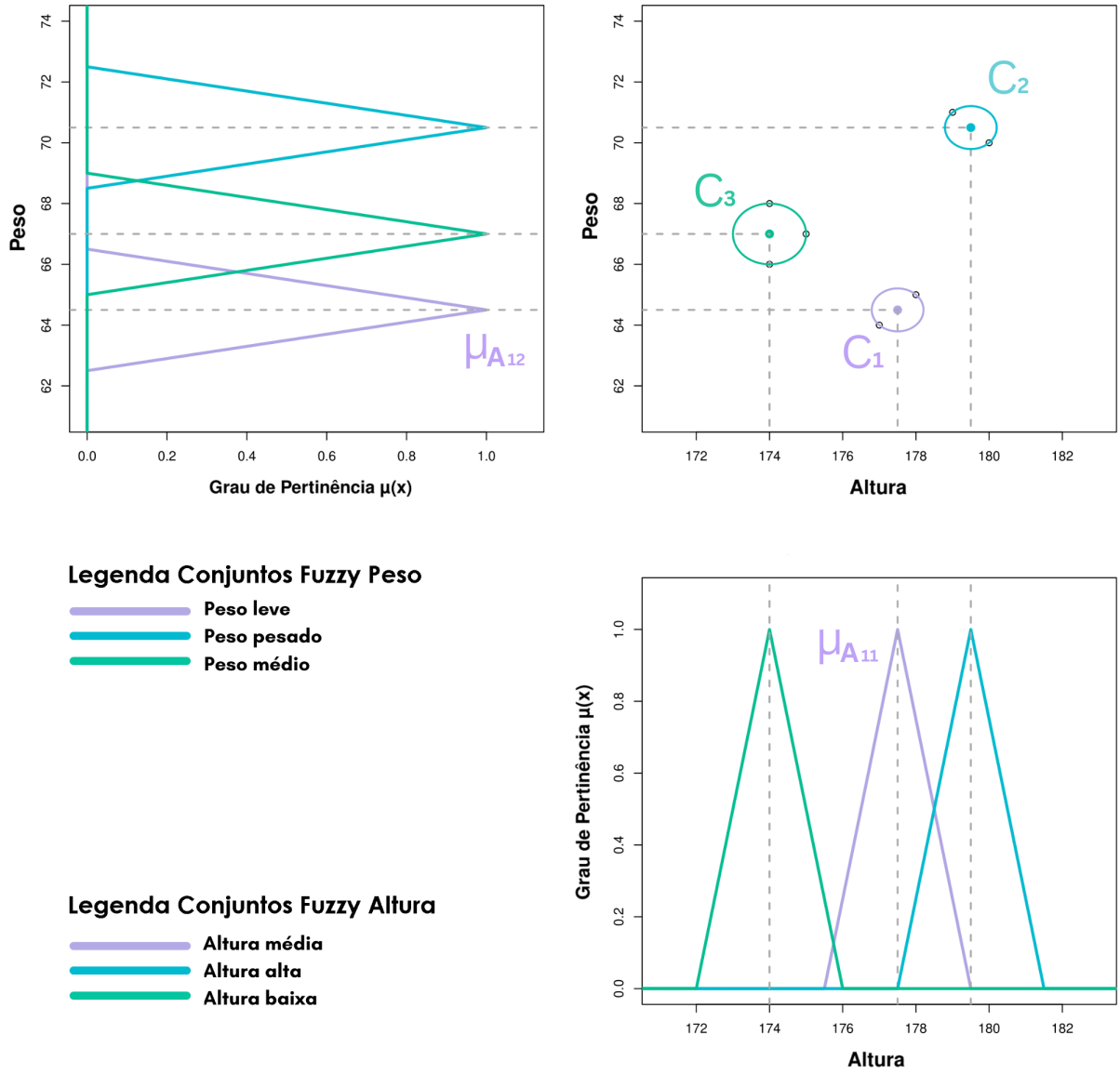
em que A_{11} é o conjunto fuzzy "ALTURA MÉDIA" (altura próxima a 177,5 cm) e A_{12} é o conjunto fuzzy "PESO LEVE" (peso próximo a 72 kg), cujas funções de pertinência são, respectivamente,

$$\mu_{A_{11}} = \mu_1(x_1) = \begin{cases} 0, & \text{se } x_i \leqslant 175{,}5 \\ \dfrac{x_i - 175{,}5}{177{,}5 - 175{,}5}, & \text{se } 175{,}5 < x_i \leqslant 177{,}5 \\ \dfrac{179{,}5 - x_i}{179{,}5 - 177{,}5}, & \text{se } 177{,}5 < x_i < 179{,}5 \\ 0, & \text{se } x_i \geqslant 179{,}5 \end{cases}, \tag{2.9}$$

$$\mu_{A_{12}} = \mu_1(x_2) = \begin{cases} 0, & \text{se } x_i \leqslant 62{,}5 \\ \dfrac{x_i - 62{,}5}{64{,}5 - 62{,}5}, & \text{se } 62{,}5 < x_i \leqslant 64{,}5 \\ \dfrac{66{,}5 - x_i}{66{,}5 - 64{,}5}, & \text{se } 64{,}5 < x_i < 66{,}5 \\ 0 & \text{se } x_i \geqslant 66{,}5 \end{cases}. \tag{2.10}$$

Na Figura 2.7 são representados os *clusters* C_1, C_2 e C_3, gerados pelo ECM aplicado no conjunto de dados da Tabela 2.2 e suas respectivas funções de pertinência.

Figura 2.7: *Clusters* gerados pelo ECM dos dados da Tabela 2.2 e suas respectivas funções de pertinência.



Fonte: Autoria própria.

Consequente das Regras Fuzzy e Estimador de Mínimos Quadrados Lineares

Para a parte consequente das regras fuzzy, no caso em que as funções são constantes $f_i(x_1, \cdots, x_q) = C_i$, $i = 1, \cdots, m$, o sistema é chamado de sistema de inferência fuzzy do tipo Takagi-Sugeno de ordem zero. O sistema é chamado de primeira ordem se $f_i(x_1, \cdots, x_q)$ são funções lineares, neste caso, o consequente da i-ésima regra DENFIS é a função linear,

$$f_i(x_1, \cdots, x_q) = b_0 + b_1 x_1 + b_2 x_2 + \cdots + b_q x_q. \tag{2.11}$$

E se as funções forem não lineares, o sistema é chamado de ordem superior (KASABOV; SONG, 2002). Na versão on-line do DENFIS é utilizado o sistema de inferência fuzzy do tipo Takagi-Sugeno de primeira ordem.

Para obter tais funções a partir de um conjunto de dados composto por p pares da forma $\{([x_{i1}, x_{i1}, \cdots, x_{iq}], y_i), i = 1,2, \cdots, p\}$, utiliza-se o Estimador de Mínimos Quadrados (EMQ) para calcular os coeficientes $b = [b_0, b_1, \cdots, b_q]^T$, aplicando a seguinte fórmula:

$$\mathbf{b} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{y},$$

em que

$$\mathbf{A} = \begin{pmatrix} 1 & x_{11} & x_{12} & \dots & x_{1q} \\ 1 & x_{21} & x_{22} & \dots & x_{2q} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{p1} & x_{p2} & \dots & x_{pq} \end{pmatrix}$$

e

$$\mathbf{y} = [y_1, y_2, \dots, y_p]^T,$$

em que o sobrescrito -1 indica a matriz inversa, e T indica a matriz ou o vetor transposto.

Saída do DENFIS

Um vetor de entrada $x = [x_1, x_2, \cdots, x_q]$ pode ativar múltiplas regras DENFIS, caso haja sobreposição dos conjuntos fuzzy antecedentes (MENDES *et al.*, 2012). Assim, o resultado da inferência y (saída do sistema) é dado pela média ponderada das regras R_i (que foram ativadas por x^0), que possuem grau de ativação,

$$\omega_i = T(\mu_i(x_1), \mu_i(x_2), \cdots, \mu_i(x_q))$$

sendo $\mu_i(x_j)$ a pertinência de x^0 no conjunto fuzzy A_{ij}, para $i = 1, \cdots, m$ e $j = 1, \cdots, q$, e T uma t-norma, como, por exemplo, o mínimo. No caso de K regras ativas, $1 \leqslant K \leqslant m$,

$$y = \frac{\sum_{i=1}^{K} \omega_i f_i(x_1, x_2, \dots, x_q)}{\sum_{i=1}^{K} \omega_i}. \tag{2.12}$$

2.7.2 Método de Agrupamento Subtrativo com Fuzzy C-Means

O *Software* R combina o método de agrupamento subtrativo, do inglês *Subtractive Clustering Method* (SBC) (CHIU, 1996), com o Fuzzy C-*Means* (BEZDEK; EHRLICH; FULL, 1984) para formar um SBRF (RIZA *et al.*, 2014). O SBC é aplicado para obter os centros dos *clusters*, determinando assim, a quantidade de *clusters*. Em seguida, as posições dos centros são otimizadas pelo algoritmo Fuzzy C-*Means*.

Método Agrupamento Subtrativo

O agrupamento subtrativo é uma extensão do método de clusterização da montanha (YAGER; FILEV, 1994) desenvolvido por Yager e Filev em 1994. O SBC é considerado um método de clusterização baseado em densidade pela sua forma de calcular os centros dos *clusters*. Seu funcionamento consiste em considerar cada ponto como um candidato a centro de *cluster*, calculando seu potencial em função das distâncias em relação a todos os demais. Um ponto é considerado de alto potencial se possuir muitos vizinhos próximos. Para sua aplicação, o conjunto de treinamento é separado em grupos de acordo com suas respectivas classes da variável de saída. Dessa forma, para extrair o conjunto de regras que permite prever cada classe, o SBC é aplicado em cada grupo separadamente (CHIU, 1996).

Considere o conjunto de dados $\{x_1, x_2, \cdots, x_n\}$ de uma classe específica, em que x_i é um vetor das variáveis de entrada no espaço em questão. Cada x_i é considerado um candidato a centro de *cluster*, e a medida do seu potencial P_i de ser, de fato, um centro é definida como

$$P_i = \sum_{j=1}^{n} e^{-\alpha \cdot d(x_i, x_j)^2}, \tag{2.13}$$

em que $\alpha = \dfrac{4}{r_a^2}$, sendo r_a uma constante positiva e d denota a distância euclidiana. A constante r_a é um raio normalizado que define uma vizinhança, pontos fora desse raio possuem pouca influência no potencial de um ponto em questão (CHIU, 1996).

O primeiro centro de *cluster* x_1^* é o ponto de maior potencial P_1^* dentre os n pontos do conjunto de dados. Após a sua escolha, o potencial de cada ponto x_i é reduzido de acordo com a sua proximidade a x_1^* pela Equação (2.14). Dessa forma, pontos próximos ao primeiro centro de *cluster* terão seu potencial mais reduzido em relação a pontos mais distantes e, portanto, será improvável que sejam escolhidos como o próximo centro (CHIU, 1996).

$$p_i \leftarrow p_i - p_1^* e^{-\beta \cdot d(x_i, x_1^*)^2}, \tag{2.14}$$

em que $\beta = \dfrac{4}{r_b^2}$, sendo r_b uma constante positiva definida como o raio que limita a vizinhança dos pontos que terão reduções em seu potencial.

Para evitar a obtenção de centros de *cluster* muito próximos, tipicamente escolhemos $r_b = 1{,}25 \cdot r_a$. Após todos os pontos terem seu potencial reduzido pela Equação (2.14), é escolhido o ponto com o maior potencial restante como o segundo centro. O processo se repete, reduzindo ainda mais o potencial de cada ponto em função de sua distância ao segundo centro. De forma geral, após obter o k-ésimo centro de *cluster*, o potencial dos outros pontos são atualizados pela fórmula

$$P_i \leftarrow P_i - P_k^* e^{-\beta \cdot d(x_i, x_k^*)^2},$$

sendo x_k^* a posição do centro do k-ésimo *cluster* e P_k^* seu potencial. Os centros de novos *clusters* e a redução dos potenciais são calculados até que o potencial restante de todos os pontos fique abaixo de uma determinada fração do potencial do primeiro centro. Em geral, utiliza-se $P_i^* < 0{,}15 \cdot P_1^*$ como critério de parada (CHIU, 1996).

Para ilustrar a criação de *clusters* utilizando o SBC, considere o conjunto de dados da Tabela 2.2, o qual contém as variáveis altura (cm) e peso (kg) de 7 atletas. Na Tabela 2.4 são apresentados os valores da Tabela 2.2 normalizados a serem utilizados neste exemplo.

Tabela 2.4: Dados normalizados de altura e peso das atletas da Tabela 2.2 para exemplo do SBC.

Atleta	Altura (Norm.)	Peso (Norm.)
A_1	0,9741	0,0000
A_2	0,9914	0,0603
A_3	0,9483	0,0172
A_4	0,9828	0,0086
A_5	0,9483	0,0345
A_6	1,0000	0,0517
A_7	0,9569	0,0259

Fonte: Autoria própria.

Apresentam-se, a seguir, os cálculos dos *clusters* para os dados da Tabela 2.4, obtidos utilizando o algoritmo do SBC de maneira iterativa. Neste exemplo, foi utilizado o raio $r_a = 0{,}25$, visando a construção de três *clusters* como calculado utilizando o ECM.

Inicialmente é calculado o valor de α:

$$\alpha = \frac{4}{r_a^2} = \frac{4}{0{,}25^2} = 64.$$

Em seguida, é calculado o potencial P_i^* de todos os n pontos, ou seja, das 7 atletas do conjunto de dados. Na Tabela 2.5 são apresentados os valores das distâncias euclidianas ao quadrado do primeiro ponto (A_1) aos pontos restantes e as influências dos outros pontos a A_1.

Tabela 2.5: Cálculo das distâncias e da influência exponencial em relação ao atleta A_1.

Par (A_1, A_j)	Distância ao Quadrado (d^2)	Influência ($e^{-64 \cdot d^2}$)
(A_1, A_1)	0,00000	1,0000
(A_1, A_2)	0,00393	0,7773
(A_1, A_3)	0,00096	0,9403
(A_1, A_4)	0,00014	0,9904
(A_1, A_5)	0,00185	0,8880
(A_1, A_6)	0,00334	0,8073
(A_1, A_7)	0,00096	0,9400
Soma (P_1)		**6,3435**

Fonte: Autoria própria.

Dessa forma, pela Tabela 2.5 e pela Equação (2.13), é obtido o potencial de A_1 ser o primeiro centro de *cluster*: $P_1^* = 6{,}5317$. Esse cálculo é realizado para calcular o potencial de todas os próximos 6 pontos do conjunto de dados, obtendo assim a Tabela 2.6, na qual é apresentada o valor do potencial de cada atleta.

Tabela 2.6: Potencial (P_i) calculado para cada atleta (candidato a centro de cluster).

Atleta	Potencial (P_i)
A_1	6,343013
A_2	6,104202
A_3	6,403294
A_4	6,450585
A_5	6,424909
A_6	6,127341
A_7	6,570652

Fonte: Autoria própria.

Observe que na Tabela 2.6 o ponto $A_7 = (175, 67)$ apresenta maior potencial. Sendo assim, este é escolhido como o primeiro centro de *cluster*. Em seguida, é calculado o valor de β:

$$\beta = \frac{4}{r_b^2} = \frac{4}{0{,}3125^2} = 40{,}96.$$

sendo $r_b = 1{,}25 \cdot r_a$ (CHIU, 1996). A próxima iteração consiste em recalcular os potenciais de cada ponto após a escolha de A_7 como primeiro centro, nesta etapa é utilizada a Equação (2.14), dessa forma:

$$p_i \leftarrow p_i - p_7^* e^{-40{,}96 \cdot d(x_i, x_7^*)^2},$$

em que $p_7^* = 6{,}570652$ e $x_7^* = (0{,}9569\ ,\ 0{,}0259)$. Assim, o segundo potencial de cada ponto é apresentado na Tabela 2.7.

Tabela 2.7: Potenciais recalculados após a escolha do primeiro centro (A_7).

Atleta	Potencial Novo (P_i^{novo})
A_1	0,0274
A_2	0,1422
A_3	-0,1272
A_4	0,1359
A_5	-0,1060
A_6	0,2018
A_7	0,0000

Fonte: Autoria própria.

Observe que a atleta $A_6 = (180, 70)$ possui o maior potencial restante, sendo escolhido como o segundo centro. As próximas iterações são calculadas de maneira análoga, até que o critério de parada seja atingido. Em geral, utiliza-se $P_i^* < 0{,}15 \cdot P_7^*$ como critério de parada. Ao final, os centros (não normalizados) dos *clusters* obtidos pelo SBC são exibidos na Tabela 2.8.

Tabela 2.8: Centros (não normalizados) dos *clusters* finais obtidos pelo método SBC.

Centro	Altura (cm)	Peso (kg)
C_1	175	67
C_2	180	70
C_3	177	64

O SBC determina os centros dos *clusters*, entretanto ao final de seu algoritmo, não necessariamente todos os pontos irão pertencer a algum *cluster*. Uma alternativa para englobar todos os pontos nos *clusters* gerados é calcular as distâncias de cada ponto aos centros dos agrupamentos e cada ponto é atribuído ao *cluster* mais próximo. Ao raio de cada *cluster* é atribuido a maior distância entre o centro do *cluster* e os pontos que lhe pertencem. No exemplo apresentado anteriormente, os *clusters* finais são exibidos na Tabela 2.9.

Tabela 2.9: *Clusters* resultantes da aplicação do algoritmo SBC aos dados da Tabela 2.2.

Cluster	Centro	Raio	Atletas	Descrição
C_1	$(175, 67)$	1,414214	x_3, x_5, x_7	Atletas baixas e de peso médio
C_2	$(180, 70)$	1,414214	x_2, x_6	Atletas altas e pesadas
C_3	$(177, 64)$	1,414214	x_1, x_4	Atletas de altura média e leves

O SBRF gerado pelo método SBC/FCM do pacote `frbs` (RIZA *et al.*, 2014) no *Software* R faz uso de funções de pertinência do tipo gaussiana que possuem a seguinte expressão

$$\mu_j(x_i) = \begin{cases} e^{-\dfrac{(x-c)^2}{2\sigma^2}}, & \text{se } c - \sigma \leqslant x \leqslant c + \sigma \ , \\ 0, & \text{caso contrário} \end{cases}$$

sendo $c \in \mathbb{R}$ o centro e $\sigma \in \mathbb{R}$ o raio do *cluster* j.

Para ilustrar a criação do antecedente das regras fuzzy e das funções de pertinência, considere os *clusters* da Tabela 2.9 criados pela aplicação do SBC no conjunto de dados da Tabela 2.2. O *cluster* C_1, de centro $c_1 = (175\ ,\ 67)$ e raio $r = 1{,}414$, gera a seguinte regra fuzzy 1,

$$\text{SE altura é } A_{11} \text{ E peso é } A_{12} \text{ ENTÃO } y_1 = f_1(x_1, \cdots, x_q),$$

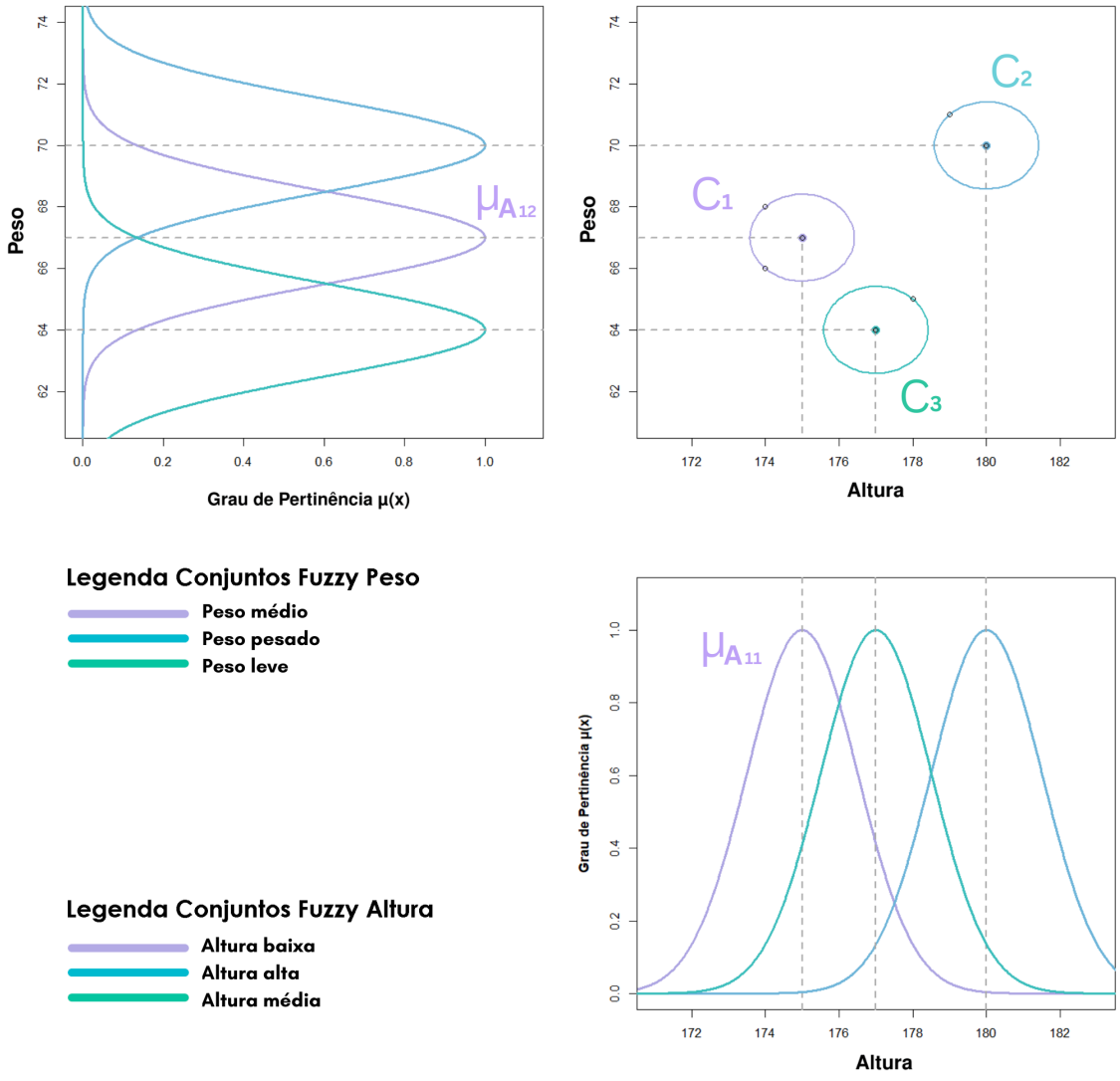
em que A_{11} é o conjunto fuzzy "ATLETA BAIXA" (altura próxima a 175 cm) e A_{12} é o conjunto fuzzy "PESO MÉDIO" (peso próximo a 67 kg), cujas funções de pertinência são, respectivamente,

$$\mu_{A_{11}} = \mu_1(x_1) = \begin{cases} e^{-\dfrac{(x-175)^2}{2 \cdot 1{,}414^2}}, & \text{se } 173{,}586 \leqslant x \leqslant 176{,}414 \\ 0, & \text{caso contrário} \end{cases}, \tag{2.15}$$

$$\mu_{A_{12}} = \mu_1(x_2) = \begin{cases} e^{-\dfrac{(x-67)^2}{2 \cdot 1{,}414^2}}, & \text{se } 65{,}586 \leqslant x \leqslant 68{,}414 \\ 0, & \text{caso contrário} \end{cases}. \tag{2.16}$$

Na Figura 2.8 são representados os *clusters* C_1, C_2 e C_3, gerados pelo SBC aplicado no conjunto de dados da Tabela 2.2 e suas respectivas funções de pertinência.

Figura 2.8: *Clusters* gerados pelo SBC dos dados da Tabela 2.2 e suas respectivas funções de pertinência.



Fonte: Autoria própria.

O método SBC/FCM utilizado pelo pacote `frbs` (RIZA *et al.*, 2014) no *Software* R emprega o SBC para determinar os centros iniciais dos *clusters*, como ilustrado no exemplo anterior, e em seguida, otimiza os centros dos *clusters* através do algoritmo Fuzzy C-*Means*, detalhado na seção a seguir.

Fuzzy C-*Means*

O Fuzzy C-*Means* (FCM) (BEZDEK; EHRLICH; FULL, 1984) é um método de clusterização que utiliza uma heurística de ponto-fixo (MEYER *et al.*, 2024). Seu algoritmo não determina a quantidade inicial de *clusters*, essa informação é dada pelo usuário ou por outro método de clusterização, como no modelo do pacote `frbs` do *Software* R (RIZA *et al.*, 2014), que combina SBC com o método Fuzzy C-*Means*. Para sua compreensão, é necessária a seguinte definição obtida em (BEZDEK; EHRLICH; FULL, 1984).

Definição 2.14. *Dado um conjunto de dados* $X = \{x_1, x_2, \cdots, x_n\} \subset R^q$*, uma c-partição fuzzy de* X *é uma matriz* $U_{c\times n}$*,*

$$U = \begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ u_{21} & u_{22} & \cdots & u_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ u_{c1} & u_{c2} & \cdots & u_{cn} \end{bmatrix},$$

onde $u_{ik} \in U$ *é o grau de pertinência da observação* x_k *no cluster* i*, que satisfaz as seguintes equações:*

$$\sum_{i=1}^{c} u_{ik} = 1, \quad \forall k \in \{1, \cdots, n\}, \tag{2.17}$$

$$0 < \sum_{k=1}^{n} u_{ik} < n, \quad \forall i \in \{1, \cdots, c\}. \tag{2.18}$$

Observe que a Equação (2.17) diz que a soma das pertinências de x_k nos c *clusters* é igual a 1, e a Equação (2.18) se refere a soma das pertinências de todas as n observações em um único *cluster* i, onde a soma precisa ser maior que 0 para que não exista *cluster* vazio e a soma é menor que n, para que todas as observações não pertença a um mesmo *cluster*, forçando a criação de no mínimo 2 *clusters*. O conjunto de todas as c-partições fuzzy de X é denotada por

$$M_{fc} = \{U_{c\times n} | u_{ik} \in [0,1]; \text{satisfazendo as Equações (2.17) e (2.18)}\}. \tag{2.19}$$

O algoritmo FCM foi desenvolvido com o objetivo de minimizar a seguinte equação,

$$J(U,v) = \sum_{k=1}^{n}\sum_{i=1}^{c}(u_{ik})^m d^2(x_k, v_i), \tag{2.20}$$

onde $x_k = (x_{k1}, x_{k2}, \cdots, x_{kq})$ é a k-ésima observação do conjunto de dados de q variáveis, c é a quantidade de *clusters* $(2 \leqslant c < n)$, $v = (v_1, v_2, \cdots, v_c)$ é o vetor dos centros dos clusters com $v_i = (v_{i1}, v_{i2}, \cdots, v_{ic})$ sendo o centro do *cluster* i, m é um expoente de ponderação $(1 \leqslant m < \infty)$ e $d(x_k, v_i)$ é a distância euclidiana da observação x_k no i-ésimo centro de *cluster*.

Para $m > 1$ se $x_k \neq v_j$ para todo j e k, o par $(\hat{U},\hat{v})$ pode ser localmente ótimo para o problema de minimização da Equação (2.20) somente se

$$\hat{v}_i = \frac{\sum_{k=1}^{n}(\hat{u}_{ik})^m x_k}{\sum_{k=1}^{n}(\hat{u}_{ik})^m}, \quad para \quad 1 \leqslant i \leqslant c, \tag{2.21}$$

e

$$\hat{u}_{ik} = \left(\sum_{j=1}^{c} \left(\frac{\hat{d}(x_k, v_i)}{\hat{d}(x_k, v_j)} \right)^{2/(m-1)} \right)^{-1}, \quad para \quad 1 \leqslant k \leqslant n \quad e \quad 1 \leqslant i \leqslant c \tag{2.22}$$

O algoritmo do Fuzzy C-*Means* é detalhado abaixo com base em (BEZDEK; EHRLICH; FULL, 1984). Considere os parâmetros $LMAX$, sendo a quantidade máxima de iterações em cada *cluster* C

Passo 1 Fixe os parâmetros c (quantidade de *clusters*), m (expoente de ponderação, também conhecido como *fuzzyfier*) e $U^{(0)}$ (matriz de pertinência inicial). Assim, para o passo k, com $k = 0,1,\cdots,LMAX$;

Passo 2 Calcule o centro de cada *cluster* $\hat{v}^{(k)} = \{v_1, v_2, \cdots, v_c\}$ utilizando a Equação (2.21);

Passo 3 Calcule e atualize a matriz de pertinência $\hat{U}^{(k+1)} = [\hat{u}_{ik}^{(k+1)}]$ com a Equação (2.22);

Passo 4 Compare $\hat{U}^{(k+1)}$ com $\hat{U}^{(k)}$ utilizando qualquer norma matricial conveniente. Se

$$d(\hat{U}^{(k+1)}, \hat{U}^{(k)}) < \epsilon, \tag{2.23}$$

pare. Caso contrário, defina $\hat{U}^{(k)} = \hat{U}^{(k+1)}$ e retorne ao Passo 2.

No algoritmo anterior, o parâmetro $LMAX$ quantifica o máximo de iterações em cada c no Passo 1, e ϵ o critério de parada no Passo 4. Valores geralmente utilizados para $LMAX$ e ϵ são 0.01 e 50 (BEZDEK; EHRLICH; FULL, 1984).

No próximo capítulo, são apresentadas a aplicação dos métodos de aprendizado de máquina e dos sistemas neuro-fuzzy na predição da indicação e do risco de lesões em atletas, assim como os resultados obtidos a partir dessa aplicação.

Capítulo 3

Aplicação dos Métodos de Aprendizado de Máquina na Predição de Lesões Esportivas em Atletas

"A gente tem de chorar no começo para sorrir no fim." (Marta [1], eleita seis vezes a melhor jogadora de futebol do mundo pela FIFA)

Neste capítulo são apresentados a base de dados utilizadas neste trabalho para realizar as predições lesões esportivas em atletas, bem como os procedimentos adotados para o pré-processamento dos dados. Além disso, este capítulo também conta com desenvolvimento e os resultados dos modelos de inteligência artificial aplicados à predição de lesão, divididos em duas partes, modelos de indicação de lesão e modelos de predição do risco de lesão.

3.1 Banco de Dados

O banco de dados utilizado neste trabalho foi coletado em uma pesquisa de monitoramento multivariado em grande escala realizada com atletas de futebol feminino profissional da Noruega (MIDOGLU *et al.*, 2024). A pesquisa foi realizada durante as temporadas de 2020 e 2021 com dois times da Liga de elite do futebol feminino norueguês, sendo 28 atletas do time A e 21 do time B, totalizando 49 atletas. As atletas registravam diariamente, através do aplicativo *Player Monitoring System* (pmSys) (JOHANSEN *et al.*, 2020), dados subjetivos relacionados ao seu bem-estar, como fadiga, estresse e duração do sono, além de dados relacionados ao treinamento. Todas as atletas forneceram seu consentimento para participar da pesquisa, incluindo a publicação aberta dos dados coletados. Com o objetivo de preservar o anonimato das atletas, todos os arquivos que incluiam alguma informação pessoal, como o nome da jogadora, foram renomeados com *strings* geradas aleatoriamente.

O estudo foi aprovado pela Autoridade Norueguesa de Proteção de Dados Pessoais (*Norwegian Privacy Data Protection Authority*), sob o número de referência 296155. Além de ter sido isento de consentimento adicional dos usuários, a pesquisa foi também isenta de aprovação adicional pelos Comitês Regionais de Pesquisa Médica e de Ética em Saúde na Noruega, uma vez que sua coleta de dados não envolveu informações pessoais sensíveis, como um bio-banco, dados médicos ou relacionados à saúde, ou interferiu no funcionamento normal das jogadoras (MIDOGLU *et al.*, 2024).

[1] ⟨https://www.espn.com.br/futebol/artigo/_/id/5763094/marta-se-emociona-e-cobra-novas-jogadoras-do-brasil-chorem-no-comeco-para-sorrir-no-fim-nao-vai-ter-marta-para-sempre⟩

A partir do banco de dados (MIDOGLU *et al.*, 2024), foram selecionadas as variáveis relacionadas ao bem-estar e à carga de treinamento, conforme descrito nas subseções a seguir. Também foi selecionada como variável resposta a ocorrência de lesão, sendo uma variável binária em que 1 representa lesão e 0 não lesão.

3.1.1 Variáveis de Bem-estar

No contexto de monitoramento de atletas e predição de lesões esportivas, as variáveis de bem-estar são indicadores subjetivos que refletem o estado físico e mental dos atletas. Reações cognitivas e emocionais, como dor, medo e luto, são associadas a incidência de lesões, uma vez que podem impactar negativamente o bem-estar dos atletas (IVARSSON *et al.*, 2017).

A utilização de questionários de bem-estar é comum no esporte de alto rendimento para identificar alertas e fornecer informações sobre sua capacidade de realizar o treinamento naquele dia (GALLO *et al.*, 2016). Dessa forma, os profissionais da área podem ajustar as cargas de treinamento com base no estado de bem-estar do atleta (SAW; MAIN; GASTIN, 2016b), o que pode contribuir para a prevenção de lesões. O banco de dados utilizado conta com 7 variáveis relacionadas ao bem-estar das atletas, que são coletadas diariamente através de questionários subjetivos:

- Estresse: Nível de estresse percebido pela atleta, em uma escala de 1 a 5.

  O estresse prolongado pode gerar mudanças nas funções neurológicas do cérebro, ocasionando uma diminuição na atenção e na capacidade de tomada de decisão, aumentando o risco de lesões (IVARSSON *et al.*, 2017).

- Fadiga: Nível de cansaço/fadiga percebido pela atleta, em uma escala de 1 a 5.

  Em um estudo realizado com jogadores de futebol profissional, foram reportadas correlações significativas (r = -0,31 a -0,56) entre fadiga autorrelatada e métricas de desempenho físico, como distância total em corridas de alta intensidade (THORPE *et al.*, 2015).

- Humor: Estado emocional geral relatado pela atleta, em uma escala de 1 a 5.

  Em um estudo realizado com atletas de futebol americano e *rugby*, foi observado que atletas que relataram mais tensão e ansiedade se lesionaram mais, enquanto atletas que relataram um estado de humor negativo geral tiveram lesões mais graves (LAVALLÉE; FLINT, 1996).

- Prontidão para Treinar: Nível de prontidão para treinar percebido pela atleta em uma escala de 1 a 10.

  O nível de prontidão para treinar também pode ser útil para monitorar o bem-estar agudo do atleta, uma vez que o estado psicológico influencia o desempenho (SAW; MAIN; GASTIN, 2016b).

- Duração do Sono: Quantidade de horas de sono na noite anterior relatada pela atleta.

- Qualidade do Sono: Qualidade do sono na noite anterior relatada pela atleta, em uma escala de 1 a 5.

- Dor Muscular: Nível de dor muscular percebido pela atleta, em uma escala de 1 a 5.

  A dor muscular e o sono podem ser indicadores para a predição de lesão. Em um estudo realizado com atletas de voleibol da Primeira Divisão da Associação Atlética Colegiada Nacional dos Estados Unidos, foi observado que, comparado com dias sem lesão, a dor

muscular, a qualidade e a duração do sono foram significativamente piores no dia anterior à ocorrência de uma lesão (HARALDSDOTTIR *et al.*, 2021).

A Tabela 3.1 apresenta um resumo das variáveis relacionadas ao bem-estar, além de suas descrições e fundamentação teórica.

Tabela 3.1: Variáveis de bem-estar.

Variáveis	**Descrição**	**Fundamentação Teórica**
Fadiga	Fadiga (1-5)	(GALLO *et al.*, 2016) (THORPE *et al.*, 2015)
Humor	Humor (1-5)	(LAVALLÉE; FLINT, 1996) (HARALDSDOTTIR *et al.*, 2021)
Prontidão para Treinar	Prontidão para Treinar (1-10)	(SAW; MAIN; GASTIN, 2016a)
Duração do Sono	Duração do sono na noite anterior em horas	(ROBERTS; TEO; WARMINGTON, 2019) (WALSH *et al.*, 2021)
Qualidade do Sono	Qualidade do sono na noite anterior (1-5)	(SAW; MAIN; GASTIN, 2016a)
Dor	Dor muscular (1-5)	(HARALDSDOTTIR *et al.*, 2021)
Estresse	Estresse (1-5)	(IVARSSON *et al.*, 2017) (SAW; MAIN; GASTIN, 2016a)

Fonte: Autoria própria.

3.1.2 Variáveis de Carga de Treinamento

Em sessões de treino de atletas de alto rendimento, a distribuição adequada da carga de treinamento é essencial para maximizar adaptações positivas e minimizar as negativas, uma vez que treinar demais pode resultar em fadiga excessiva e, consequentemente, em um maior risco de lesão, enquanto treinar pouco pode influenciar o desempenho esportivo do atleta (GABBETT, 2020). Dessa forma, surge a necessidade de monitorar a carga de treinamento dos atletas para otimizar o desempenho e prevenir lesões.

A escala PSE desenvolvida por Borg (BORG, 1998) é uma forma válida de monitorar o treinamento em diversos tipos de esportes e exercícios (FOSTER *et al.*, 2001), além de se destacar pela praticidade na coleta dos dados. O PSE é uma ferramenta subjetiva utilizada para quantificar a intensidade de um exercício, considerando a percepção do atleta em uma escala de 1 a 10. Em um estudo de previsão de lesões realizado por Gabbett (GABBETT, 2010), utilizando o PSE da sessão, foi mostrado que jogadores que excederam o limite de carga de treinamento semanal tiveram 70 vezes mais probabilidade se lesionarem.

O banco de dados utilizado conta com 9 variáveis relacionadas à carga de treinamento, calculadas a partir do PSE da sessão, que eram coletadas após cada sessão de treinamento ou esforço físico realizado pelas atletas (MIDOGLU *et al.*, 2024):

- Percepção Subjetiva de Esforço da Sessão (PSEs): Carga de uma sessão de treinamento baseada na duração da sessão e no PSE reportado em uma escala de 1 a 10 (BORG, 1998), dada por:

$$\text{PSEs} = \text{duração} \cdot \text{PSE}.$$

- Carga Diária (CD): Soma dos PSEs relatados durante o dia, dada por:

$$\text{CD} = \sum \text{PSEs}.$$

- Carga Semanal (CS): Soma dos PSEs relatados nos últimos 7 dias, ou seja, a soma da Carga Diária dos últimos 7 dias, dada por:

$$\text{CS} = \sum_{i=n}^{n+6} \text{CD}_i,$$

sendo CD_i a Carga Diária do i-ésimo dia e n o dia atual.

O monitoramento da carga realizado com frequência, como diárias ou semanais, possibilitam ajustes imediatos nas cargas de treino e competição quando necessário. Grandes aumentos na carga semana a semana, ou variações bruscas, podem aumentar o risco de lesão por não permitir adaptações fisiológicas adequadas (SOLIGARD *et al.*, 2016).

- Carga Aguda de Treinamento (CAT): Nível atual de fadiga, calculado como a média do PSEs nos últimos 7 dias, dada por:

$$\text{CAT} = \frac{1}{7} \cdot \sum_{i=n}^{n+6} \text{CD}_i,$$

sendo CD_i a Carga Diária do i-ésimo dia e n o dia atual.

- Carga Crônica de Treinamento 28 (CCT28): Dose acumulativa de treinamento que aumenta ao longo dos últimos 28 dias, dada por:

$$\text{CCT28} = \frac{1}{28} \cdot \sum_{i=n}^{n+27} \text{CD}_i,$$

sendo CD_i a Carga Diária do i-ésimo dia e n o dia atual.

- Carga Crônica de Treinamento 42 (CCT42): Dose acumulativa de treinamento que aumenta ao longo dos últimos 42 dias, dada por:

$$\text{CCT42} = \frac{1}{42} \cdot \sum_{i=n}^{n+41} \text{CD}_i,$$

sendo CD_i a Carga Diária do i-ésimo dia e n o dia atual.

- Relação de Carga Aguda-Crônica 28 (RAC): Indica se um atleta está em um estado bem preparado ou em um risco de se machucar, calculada por:

$$\text{RAC} = \frac{\text{CAT}}{\text{CCT28}}.$$

- Relação de Carga Aguda-Crônica 42 (RAC): Indica se um atleta está em um estado bem preparado ou em um risco de se machucar, calculada por:

$$\text{RAC} = \frac{\text{CAT}}{\text{CCT42}}.$$

Enquanto a carga aguda refere-se à carga de treinamento ou competição de um curto período, em geral de uma semana, a carga crônica representa a média dessa carga ao longo de um período mais longo, geralmente quatro ou seis semana. Em um estudo realizado por Windt e Gabbett (WINDT; GABBETT, 2017) foi demonstrado que a relação entre essas duas cargas está fortemente associada ao risco de lesão em atletas. Atletas que possuiam uma Relação de Carga Aguda-Crônica (RAC) entre 0,8 e 1,3 apresentaram uma baixa probabilidade de lesão (menor que 10%). Enquanto atletas que apresentaram um RAC maior que 1,5 (ou seja, a carga na semana mais recente é 1,5 maior do que a carga nas últimas 4 semanas) tiveram mais que o dobro da probabilidade de se lesionarem (WINDT; GABBETT, 2017).

- Monotonia: Variação do treinamento semanal, calculada como a CAT dividida pelo desvio padrão das cargas diárias dos últimos 7 dias, dada por:

$$\text{Monotonia} = \frac{\text{CAT}}{\sigma},$$

sendo σ o desvio padrão das cargas diárias dos últimos 7 dias.

- Tensão do Treinamento: Estresse geral do treinamento dos últimos 7 dias, calculada como a Carga Semanal multiplicada pela Monotonia, dada por:

$$\text{Tensão} = \text{CS} \cdot \text{Monotonia}.$$

Enquanto a monotonia indica a forma como a carga de treinamento do atleta é distribuída ao longo da semana, a tensão reflete o estresse geral daquela semana. Ambas variáveis estão associadas a desfechos adversos do treinamento, como lesões (FOSTER *et al.*, 2001).

Um treinamento intenso com pouca variação de carga na semana (alta média semanal e baixo desvio padrão) indica alta monotonia. Enquanto na variável tensão, um alto valor pode indicar que o atleta teve uma semana muito pesada (alta carga semanal) e muito repetitiva (alta monotonia), considerado um risco para a saúde do atleta, uma vez que não teve variação de estímulo e nem tempo de descanso para se recuperar de maneira adequada.

Em um estudo realizado com atletas universitárias de futebol feminino, mostrou-se que a alta monotonia está associada a um aumento na incidência de lesões e baixo desempenho esportivo (PUTLUR *et al.*, 2004). Em um outro estudo, 77% das lesões foram associadas a um pico prévio na monotonia, enquanto 89% das lesões puderam ser explicadas por um pico precedente na tensão do treinamento (FOSTER, 1998).

Na Tabela 3.2, é apresentado um resumo das variáveis relacionadas à carga de treinamento, além de suas descrições, cálculos e fundamentação teórica.

Tabela 3.2: Variáveis de carga de treinamento.

Variáveis	**Descrição**	**Cálculo**	**Fundamentação Teórica**
Classificação do Esforço Percebido por sessão (PSEs)	A carga de uma sessão de treinamento baseada na duração da sessão e no CEP reportado em uma escala de 1 a 10	$\text{duração} \cdot \text{PSE}$	(IMPELLIZZERI *et al.*, 2004) (FOSTER *et al.*, 2001)
Carga Diária (CD)	A soma dos PSEs relatados durante o dia	$\sum \text{PSEs}$	(SOLIGARD *et al.*, 2016)
Carga Semanal (CS)	A soma dos PSEs relatados nos últimos 7 dias	$\sum \text{PSEs}$	(JASPERS *et al.*, 2018) (SOLIGARD *et al.*, 2016)
Carga Aguda de Treinamento (CAT)	O nível atual de fadiga (média do CD nos últimos 7 dias)	$\frac{1}{7} \cdot \sum_{i=n}^{n+6} \text{CD}_i$	(GABBETT, 2016) (SOLIGARD *et al.*, 2016)
Carga Crônica de Treinamento 28 (CCT28)	A dose acumulativa de treinamento que aumenta ao longo dos últimos 28 dias	$\frac{1}{28} \cdot \sum_{i=n}^{n+27} \text{CD}_i$	(GABBETT, 2016) (SOLIGARD *et al.*, 2016)
Carga Crônica de Treinamento 42 (CCT42)	A dose acumulativa de treinamento que aumenta ao longo dos últimos 42 dias	$\frac{1}{42} \cdot \sum_{i=n}^{n+41} \text{CD}_i$	(GABBETT, 2016) (SOLIGARD *et al.*, 2016)
Relação de Carga Aguda-Crônica 28 (RAC28)	Uma indicação de se um atleta está em um estado bem preparado ou em um risco de se machucar	$\frac{\text{CCT28}}{\text{CAT}}$	(GABBETT, 2016) (HULIN *et al.*, 2016) (WINDT; GABBETT, 2017)
Relação de Carga Aguda-Crônica 42 (RAC42)	Uma indicação de se um atleta está em um estado bem preparado ou em um risco de se machucar	$\frac{\text{CCT42}}{\text{CAT}}$	(GABBETT, 2016) (HULIN *et al.*, 2016) (WINDT; GABBETT, 2017)
Monotonia	Variação do treinamento semanal (CAT dividido pelo desvio padrão das cargas diárias dos últimos 7 dias)	$\frac{\text{CAT}}{\sigma}$	(FOSTER, 1998) (FOSTER *et al.*, 2001) (PUTLUR *et al.*, 2004)
Tensão	Estresse geral do treinamento dos últimos 7 dias	$\text{CS} \cdot \text{Monotonia}$	(FOSTER, 1998) (FOSTER *et al.*, 2001) (PUTLUR *et al.*, 2004)

Fonte: Autoria própria.

Na Tabela 3.3, é exemplificado o cálculo das variáveis relacionadas à carga de treinamento a partir dos dados brutos inseridos pelas atletas no aplicativo pmSys. No cálculo da Carga

Crônica de Treinamento 28 (CCT28) e no da Carga Crônica de Treinamento 42 (CCT42), foram considerados os valores de Carga Diária (CD) nos dias anteriores ao domingo na tabela como sendo zero.

Tabela 3.3: Exemplo de monitoramento semanal da carga de treinamento.

Dia	**Atividade**	**PSE**	**Duração (min)**	**Carga Diária**
Domingo	Corrida	8	35	280
Segunda	Treinamento em campo	7	70	490
Terça	Descanso	0	0	0
Quarta	Aquecimento para jogo	5	41	
	Jogo	9	45	610
Quinta	Treinamento em campo	7	50	350
Sexta	Musculação	8	75	390
Sábado	Treinamento em campo	6	40	240
	Carga Semanal			2.450
	Carga Aguda			350
	Carga Crônica 28			87,5
	Carga Crônica 42			58,33
	Relação de Carga Aguda Crônica 28			4
	Relação de Carga Aguda Crônica 42			6
	Monotonia			1,87
	Tensão			4.581,5

Fonte: Autoria própria.

Destaca-se que a Carga Diária de quarta-feira é a soma do PSEs do aquecimento para o jogo (205) e do jogo em si (405), totalizando 610.

3.1.3 Pré-Processamento dos Dados

O banco de dados inicial contava com 11.503 registros de atletas sem lesão e 173 registros de atletas lesionadas, totalizando 11.676 observações. Durante a leitura dos dados, foi possível observar que algumas atletas inseriram duas ou mais lesões em um mesmo dia. Como o objetivo do trabalho é a predição da ocorrência de lesão, foi necessário colocar cada ocorrência de lesão em uma linha, ou seja, se uma atleta inseriu duas lesões em um mesmo dia, o registro foi duplicado, colocando cada lesão em uma linha. Após esse procedimento, o *dataset* ficou com 308 registros de atletas lesionadas, totalizando 11.811 observações.

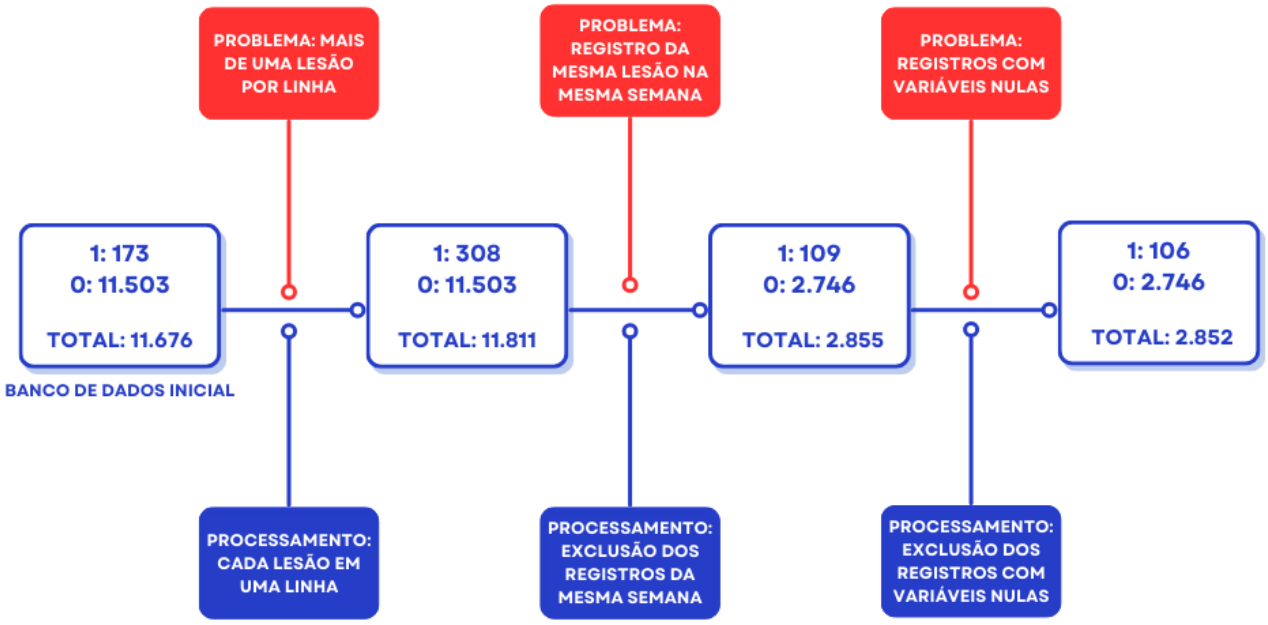
Verificou-se, ainda, que algumas atletas registravam a mesma lesão em dias consecutivos, por ainda estarem com o trauma. Como o objetivo do trabalho é prever a lesão utilizando o histórico de registros da atleta, ou seja, os dados anteriores à lesão, tal fato poderia influenciar negativamente na construção do modelo. Dessa forma, foi necessário excluir registros de uma mesma lesão em um período de 7 dias, obtendo assim 109 observações de atletas lesionadas. Para seguir o padrão, o mesmo procedimento foi realizado nos registros de atletas sem lesão, excluindo registros em um período de 7 dias, obtendo assim 2.746 observações de atletas sem lesão. Ao final, também foram excluídos três registros que possuiam variáveis nulas. Assim, o *dataset* final ficou com 2.852 observações totais, cuja distribuição entre "lesão" e "não lesão" é evidenciada na Tabela 3.4.

Na Figura 3.1, é ilustrado um diagrama com o resumo do pré-processamento realizado no conjunto de dados.

Tabela 3.4: Distribuição das classes no conjunto de dados.

Classe 0 (Sem Lesão)	Classe 1 (Com Lesão)
2746	106

Figura 3.1: Diagrama do pré-processamento realizado no conjunto de dados.



Fonte: Autoria própria.

3.2 Resultados Obtidos nos Modelos de Aprendizado de Máquina para Indicação de Lesão

Nesta seção, são apresentados os resultados obtidos pelos métodos de aprendizado de máquina para predição da indicação de lesão em atletas, sendo divididos em duas subseções: os resultados obtidos pela árvore de decisão e os resultados obtidos pela floresta aleatória. Os algoritmos foram desenvolvidos utilizando os pacotes `rpart.plot` e `randomForest` do *Software* R (THERNEAU; ATKINSON, 2024; LIAW; WIENER, 2002).

As árvores e florestas foram treinadas com 70% dos dados e testadas com os 30% restantes. O conjunto de treino contava com 1998 observações, sendo 1923 registros sem lesão e 75 com lesão. O conjunto de teste contava com 854 observações, sendo 823 registros sem lesão e 31 com lesão.

As variáveis de entrada utilizadas foram: “Carga Diária”, “Carga Aguda de Treinamento”, “Carga Semanal”, “Monotonia”, “Tensão do Treinamento”, “Carga Crônica de Treinamento 28”, “Carga Crônica de Treinamento 42”, “Fadiga”, “Prontidão para Treinar”, “Duração do Sono” e “Qualidade do Sono”.

3.2.1 Resultados Obtidos pela Árvore de Decisão

O modelo da Árvore de Decisão foi desenvolvido utilizando o método de validação cruzada do tipo k-*fold* com $k = 5$ (quantidade de *folds* padrão) para definir o valor do hiperparâmetro *cp* (Parâmetro de Complexidade) que resultasse em maior acurácia. Na Tabela 3.5 são apresentados os valores de *cp* testados e os resultados obtidos para cada um deles, ordenados pela acurácia média.

Tabela 3.5: Resultados da Validação Cruzada por Parâmetro de Complexidade (cp) para a Árvore de Decisão.

cp	**Accuracy**
0.006666667	0.9674637
0.043333333	0.9664649
0.106666667	0.9624624

O valor de *cp* que apresentou a melhor acurácia média foi 0.0066, escolhido para treinar o modelo final.

A Tabela 3.6 representa a matriz de confusão do modelo de predição de indicação de lesão do modelo gerado pela Árvore de Decisão para os dados de teste, sendo as linhas os valores reais e as colunas os valores preditos.

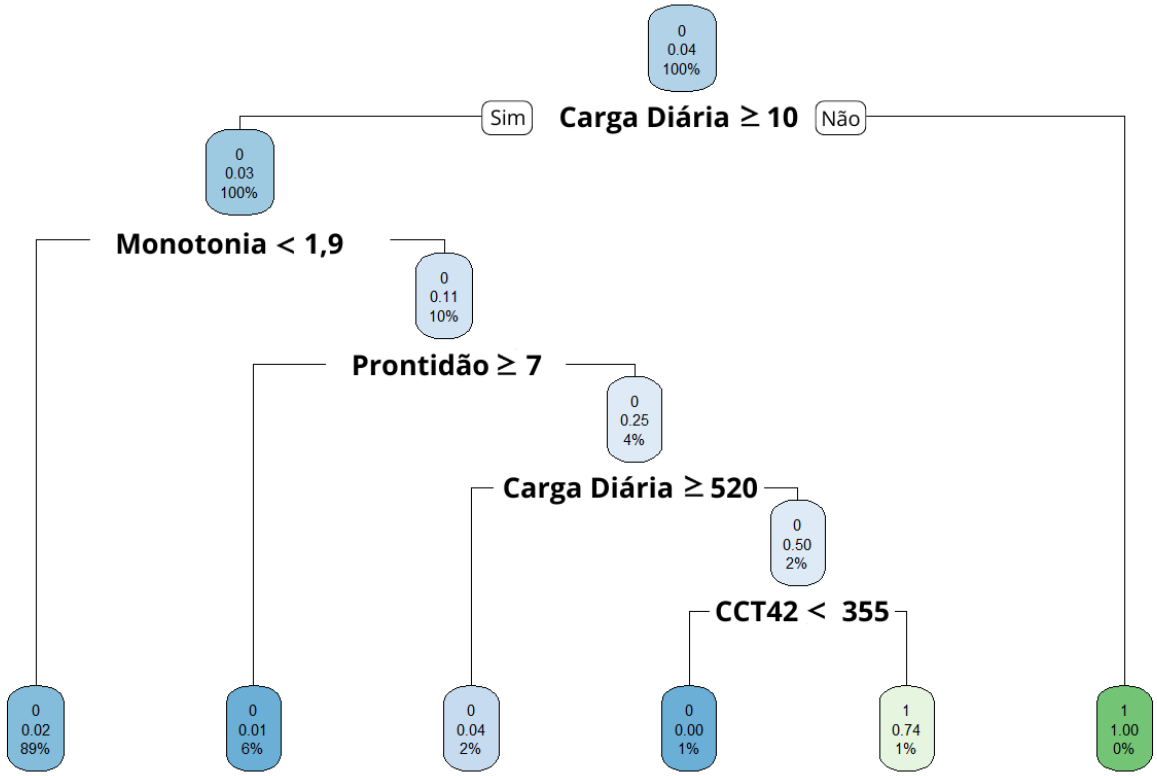
Tabela 3.6: Matriz de confusão da árvore de decisão.

	0	**1**
0	816	7
1	17	14

O modelo final obteve uma acurácia de 97,19% e sensibilidade de 45,16% no conjunto de teste.

A Figura 3.2 apresenta o diagrama da árvore de decisão obtida.

Figura 3.2: Modelo final da árvore de decisão da predição da indicação de lesão.



Fonte: *Software R*.

Nos tópicos a seguir é apresentada a estrutura da árvore ilustrada na Figura 3.2.

- A árvore gerada possui 5 nós, sendo a primeira divisão realizada pela variável "Carga Diária", em que valores menores que 10 são classificados como "Lesão" (folha verde da classe 1) e valores maiores ou iguais a 10 seguem para o próximo nó.

- No segundo nó, a variável utilizada é a "Monotonia", em que valores menores que 1,9 são classificados como "Não Lesão" (folha azul da classe 0) e valores maiores ou iguais a 1,9 seguem para o próximo nó.

- No terceiro nó, a variável utilizada é a "Prontidão para Treinar", em que valores maiores ou iguais a 7 são classificados como "Não Lesão" e valores maiores ou iguais a 7 seguem para o próximo nó.

- No quarto nó, a variável utilizada é novamente a "Carga Diária", em que valores maiores ou iguais a 520 são classificados como "Não Lesão" e valores menores que 520 seguem para o último nó.

- Por fim, no quinto nó, a variável utilizada é a "Carga Crônica 42", em que valores menores que 355 são classificados como "Não Lesão" e valores maiores ou iguais a 355 são classificados como "Lesão".

Na subseção a seguir são apresentados os resultados obtidos pela Floresta Aleatória.

3.2.2 Resultados Obtidos pela Floresta Aleatória

A Floresta Aleatória foi treinada utilizando como hiperparâmetros o número de árvores igual a 5.000 e o número de variáveis consideradas em cada divisão (*mtry*) igual a 3. A Tabela 3.7 representa a matriz de confusão do modelo de predição de indicação de lesão gerado pela Floresta Aleatória para os dados de teste, sendo as linhas os valores reais e as colunas os valores preditos.

Tabela 3.7: Matriz de confusão da floresta aleatória.

	0	**1**
0	823	0
1	11	20

O modelo obtido pela Floresta Aleatória alcançou uma acurácia de 98,71% e sensibilidade de 64,52% no conjunto de teste.

3.2.3 Comparação entre a Árvore de Decisão e a Floresta Aleatória

Na Tabela 3.8 é apresentada a comparação entre a acurácia e sensibilidade obtidas pelos dois métodos no conjunto de teste.

Tabela 3.8: Valores de acurácia e sensibilidade obtidos pela Árvore de Decisão e pela Floresta Aleatória para predição da indicação de lesão.

Método	Acurácia	Sensibilidade
Árvore de Decisão	97,19%	45,16%
Floresta Aleatória	98,71%	64,52%

Fonte: Autoria própria.

Comparando as métricas obtidas pelos dois métodos no conjunto de teste, observa-se que a Floresta Aleatória superou a Árvore de Decisão em 1,52% de acurácia e 19,36% de sensibilidade (Tabela 3.8). Tal conclusão é condizente com a teoria, uma vez que a Floresta Aleatória, no geral, obtém melhores resultados do que uma única Árvore de Decisão, pois a Floresta Aleatória é um conjunto de várias Árvores de Decisão, capaz de capturar melhor as relações entre as variáveis de entrada e a variável de saída (BREIMAN, 2001).

O modelo desenvolvido para predição da indicação de lesão permite realizar predições para novas atletas com outros valores de variáveis de entrada. Na Tabela 3.9 são apresentadas as predições da indicação ou não de lesão para 3 atletas diferentes.

Tabela 3.9: Predição do modelo de indicação de lesão para 3 novas atletas.

Variável	Atleta 1	Atleta 2
CD	930	10
CAT	250	234,29
CS	1750	1640
Monotonia	0,8	1,76
Tensão	1400	2886,4
CCT28	387,86	230,36
CCT42	375,95	250,90
Fadiga	2	5
Prontidão para Treinar	10	2
Duração do Sono	8	4
Qualidade do Sono	4	1
Predição da Indicação da Árvore de Decisão	0	0
Predição da Indicação da Floresta Aleatória	0	1

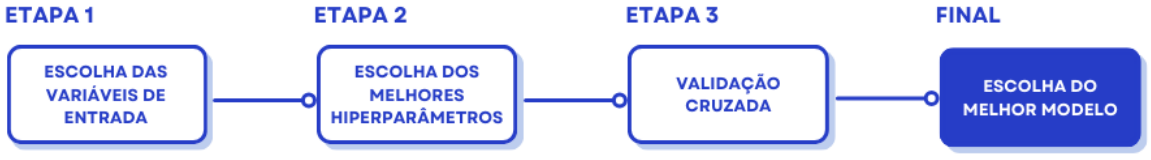
Fonte: Autoria própria.

A Atleta 1 apresenta boas condições de treinamento, com baixas valores de carga de treinamento e boas medidas de bem-estar, sendo classificada por ambos os modelos como "Não Lesão". Por outro lado, a Atleta 2 apresenta altos valores de carga de treinamento acompanhado de valores de bem-estar considerados inadequados, como baixa duração de sono (4 horas), baixa qualidade de sono (1), baixa prontidão para treinar (2) e o valor máximo de fadiga (5). Dessa forma, a Atleta 2 foi classificada pela Floresta Aleatória como "Lesão", condizente com o esperado. Em contrapartida, a Árvore de Decisão classificou a Atleta 2 como "Não Lesão", apresentando um possível erro na predição e demonstrando a superioridade da Floresta Aleatória em comparação com a Árvore de Decisão.

3.3 Resultados Obtidos nas Redes Neuro-Fuzzy para Predição do Risco de Lesão

Nesta seção, são apresentados os resultados obtidos pelos modelos de Redes Neuro-Fuzzy para predição do risco de lesão em atletas, divididos em duas subseções: os resultados obtidos pelo DENFIS e os resultados obtidos pelo SBC/FCM. Ao final, é realizada uma comparação entre os dois métodos e as predições do risco para novas atletas. Em ambos os métodos, foi adotada a mesma metodologia, dividida em três etapas, para realizar a busca pelas melhores variáveis de entrada e hiperparâmetros, conforme ilustrado na Figura 3.3.

Figura 3.3: Diagrama da metodologia.



Fonte: Autoria própria.

Conforme ilustrado na Figura 3.3, a metodologia consistiu nas seguintes etapas:

Etapa 1. Escolha das variáveis de entrada: Foram avaliadas em um *looping* todas as combinações possíveis de 9 variáveis de entrada de 4 em 4, totalizando 126 combinações. Apesar do banco de dados utilizado (MIDOGLU *et al.*, 2024) possuir 14 variáveis, considerando o custo computacional e o tempo de execução dos *loopings*, optou-se por utilizar apenas 9 variáveis, focando nas variáveis de carga de treinamento e, dentre as de bem-estar, foi utilizada a duração do sono. Dentre as variáveis de carga de treinamento, apenas a Carga Crônica 42 foi excluída da análise. Quanto aos indicadores de bem-estar, optou-se exclusivamente pela Duração do Sono, dada a sua relevância conforme a literatura (WALSH *et al.*, 2021). As 9 variáveis escolhidas foram: Carga Diária, Carga Semanal, Carga Aguda, Carga Crônica 28, Relação Carga Aguda-Crônica 28, Relação Carga Aguda-Crônica 42, Monotonia, Tensão e Duração do Sono.

Cada combinação foi avaliada nos modelos DENFIS e SBC/FCM, utilizando o conjunto de dados dividido em 70% para treinamento e 30% para teste. Inicialmente, no conjunto de treinamento havia 1.927 observações sem lesão (0) e 70 com lesão (0). Considerando o alto nível de desbalanceamento, foi aplicada a técnica SMOTE nos dados de treino, obtendo assim 1.927 linhas sem lesão e 1.120 com lesão.

Durante essa etapa também foi analisado o ponto de corte. Uma vez que no banco de dados, a variável de saída "ocorrência de lesão" é binária (0 ou 1) e os modelos de rede neuro-fuzzy realiza predições contínuas do risco de lesão, um valor entre 0 e 1, foi necessário o arredondamento das predições para o cálculo das métricas de desempenho como acurácia e sensibilidade. Seja $\hat{p}$ a predição de risco do modelo, a classificação binária é obtida por

$$\hat{y}_c = \begin{cases} 1, & \text{se } \hat{p} \geqslant c \\ 0, & \text{se } \hat{p} < c \end{cases},$$

em que c é o ponto de corte. Visto que o modelo desenvolvido possui caráter preventivo, com o intuito de considerar aquelas atletas lesionadas (1) que foram preditas com riscos intermediários abaixo de 0,5, foi avaliada uma grade de pontos de corte $c \in \{0{,}40, 0{,}30, 0{,}25, 0{,}20\}$ e, para cada c, foi recalculada a acurácia e sensibilidade no conjunto de teste. Os pontos de cortes testados são menores ou iguais a 0,5, evidenciando que se pretende obter um modelo preventivo, com finalidade de priorizar o risco de lesões nas atletas.

Os parâmetros utilizados nesta etapa foram escolhidos de maneira empírica, sendo estes:

1.1 Método DENFIS: $dthr = 0{,}05$, $maxiter = 300$, $stepsize = 0{,}03$ e $d = 3$, sendo $maxiter$ o número máximo de iterações do algoritmo e $stepsize$ o tamanho do passo.

1.2 Método SBC/FCM: $r_a = 0{,}05$, $eps.high = 0{,}15$ e $eps.low = 0{,}01$, sendo os dois últimos parâmetros utilizados no método SBC/FCM no pacote `frbs` (RIZA *et al.*, 2014) do *Software* R.

Etapa 2. Escolha dos melhores hiperparâmetros empiricamente: Em seguida foram avaliados diferentes hiperparâmetros para cada método, utilizando as variáveis escolhidas nas etapas anteriores. Nesta etapa também foi considerado os pontos de corte utilizados na etapa anterior. Os hiperparâmetros avaliados foram:

2.1 Método DENFIS:

* DTHR: Os valores do hiperparâmetro "DTHR" são apresentados na Tabela 3.10.

Tabela 3.10: Valores do hiperparâmetro "DTHR" avaliados na Etapa 2.

Valores DTHR				
0,03	0,05	0,055	0,07	0,1

* Número máximo de iterações do algoritmo (*maxiter*): Os valores do hiperparâmetro "maxiter" são apresentados na Tabela 3.11.

Tabela 3.11: Valores do hiperparâmetro "maxiter" avaliados na Etapa 2.

Valores maxiter		
100	200	300

* Tamanho do passo (*stepsize*): Os valores do hiperparâmetro "stepsize" são apresentados na Tabela 3.12.

Tabela 3.12: Valores do hiperparâmetro "stepsize" avaliados na Etapa 2.

Valores stepsize		
0,01	0,02	0,03

* d: Os valores do hiperparâmetro "d" são apresentados na Tabela 3.13.

Tabela 3.13: Valores do hiperparâmetro "d" avaliados na Etapa 2.

Valores d		
2	2,5	3

Totalizando $5 \cdot 3 \cdot 3 \cdot 3 = 135$ combinações de hiperparâmetros.

2.2 Método SBC/FCM:

* r.a: Os valores do hiperparâmetro "r.a" são apresentados na Tabela 3.14.

Tabela 3.14: Valores do hiperparâmetro "r.a" avaliados na Etapa 2.

Valores r.a							
0,01	0,015	0,02	0,025	0,03	0,035	0,04	0,045
0,048	0,05	0,053	0,055	0,058	0,06	0,065	0,07
0,08	0,085	0,09	0,1	0,15	0,2	0,25	0,3

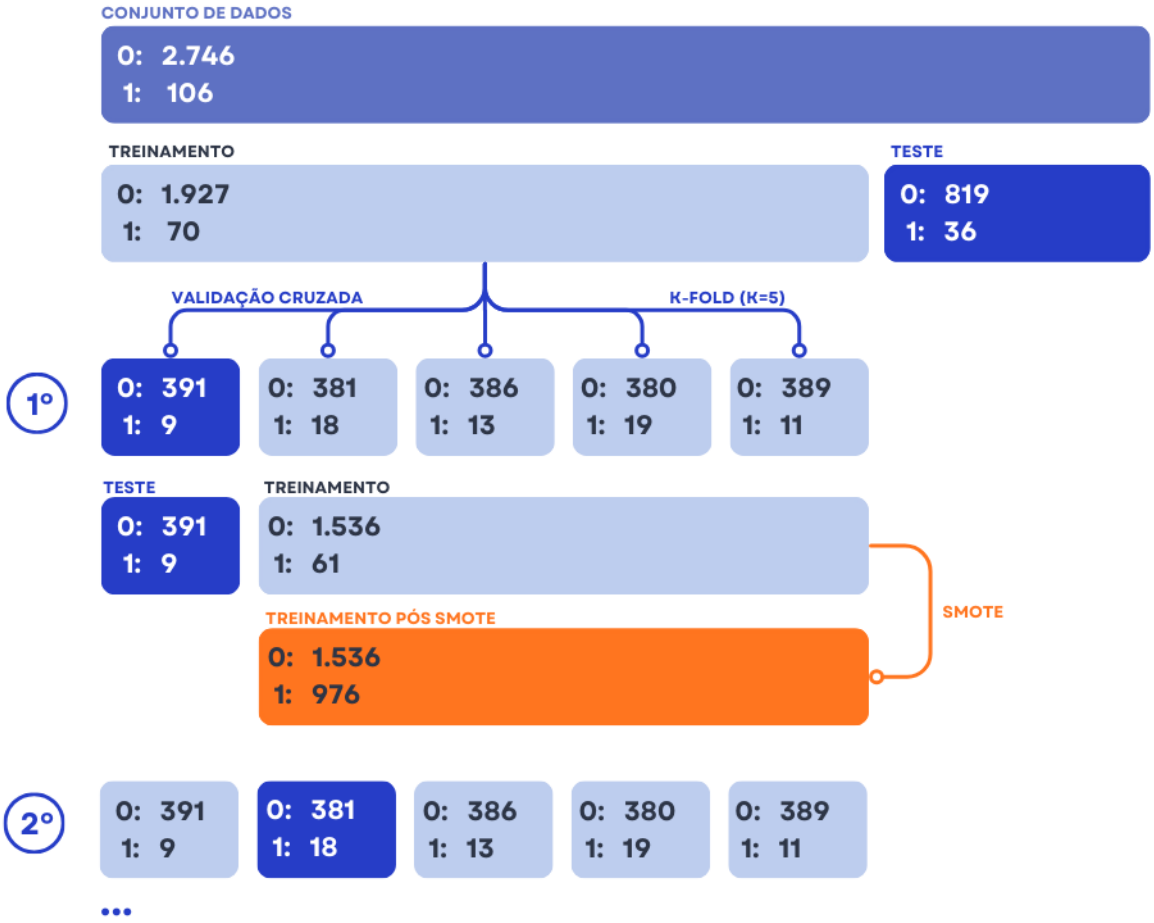
* eps.high: 0,15.
* eps.low: 0,01.

Totalizando $24 \cdot 1 \cdot 1 = 24$ combinações de hiperparâmetros.

No método SBC/FCM foi observado em testes anteriores que alterar os valores dos parâmetros "eps.high" e "eps.low" não alteravam o resultado final do modelo. Dessa forma, considerando o custo computacional, foi escolhido apenas um valor para cada um desses parâmetros.

Etapa 3. Validação cruzada: Por fim, foi realizada validação cruzada do tipo k-*fold* ($k = 5$) com os modelos que obtiveram melhor resultado na etapa anterior. Na Figura 3.4 é apresentada a ilustração do processo de validação cruzada k-*fold* utilizado neste trabalho. A escolha dos dados de cada *fold* é aleatória, se possível, mantendo a quantidade de dados entre estes, conforme o comando "createFolds" do pacote `caret` (KUHN, 2008) no *Software* R.

Figura 3.4: Processo de validação cruzada k-*fold* ($k = 5$) realizado no conjunto de treinamento.



Fonte: Autoria própria.

Inicialmente, o conjunto de dados é divido em 70% para treinamento e 30% para teste, como ilustrado na Figura 3.4. Em seguida, começa o processo de validação cruzada do tipo k-*fold*, com $k = 5$. Nesta etapa, o conjunto de treinamento é dividido em 5 subconjuntos de tamanho aproximadamente igual. A cada iteração, um subconjunto é utilizado como conjunto de teste, enquanto os 4 subconjuntos restantes são reservados para o treinamento do modelo. Na primeira iteração o conjunto de teste contava com 391 observações sem lesão e 9 com lesão, e o conjunto de treinamento com 1.536 linhas sem lesão e 61 com lesão. Dessa forma, considerando o alto desbalanceamento de classes do conjunto de treino, foi aplicada a técnica SMOTE para criação de dados sintéticos no conjunto de treinamento de cada iteração, assim como no desenvolvimento dos modelos das etapas anteriores. Com isso, na primeira iteração foi obtido um conjunto de treinamento com 1.536 linhas sem lesão e 976 com lesão, e treinado o modelo do primeiro *fold*. Esse processo é repetido 5 vezes, permitindo que cada observação seja utilizada tanto para treinamento quanto para teste. Ao final, é realizada a média das métricas sensibilidade e acurácia dos 5 modelos

gerados nesse processo, otendo uma estimativa mais robusta acerca do desempenho do modelo como ilustrado na Figura 1.1. É importante ressaltar que o modelo escolhido é aquele treinado com todos os dados de treinamento.

Em todas as etapas, os modelos foram treinados com dados normalizados. As métricas utilizadas para avaliar o desempenho dos modelos foram Sensibilidade e Acurácia.

3.3.1 Resultados Obtidos pelo DENFIS

Nesta subseção, são apresentados os resultados obtidos através do método DENFIS nas 3 Etapas apresentadas na Figura 3.3. A Tabela 3.15 apresenta as 15 combinações de variáveis de entrada com os maiores valores de sensibilidade e os seus respectivos valores de acurácia obtidos pelo método DENFIS, resultados da Etapa 1 da análise.

Tabela 3.15: Desempenho das 15 combinações de variáveis com maiores valores de sensibilidade pelo método DENFIS.

Variáveis	Ponto de corte	Métricas	
		Sensibilidade	Acurácia
Monotonia, Tensão, CCT28, RAC42	0,2	88,89%	61,05%
CAT, Monotonia, Tensão, CCT28	0,25	83,33%	66,90%
CS, Monotonia, Tensão, CCT28	0,25	83,33%	66,90%
CAT, Monotonia, Tensão, CCT28	0,2	83,33%	63,39%
CS, Monotonia, Tensão, CCT28	0,2	83,33%	63,39%
CAT, CS, Monotonia, Tensão	0,2	83,33%	35,67%
CAT, CS, Monotonia, CCT28	0,3	80,56%	74,50%
CAT, Tensão, CCT28, Duração do Sono	0,2	80,56%	72,16%
CS, Tensão, CCT28, Duração do Sono	0,2	80,56%	72,16%
CAT, Monotonia, Tensão, CCT28	0,3	80,56%	70,64%
CS, Monotonia, Tensão, CCT28	0,3	80,56%	70,64%
CAT, CS, Monotonia, CCT28	0,25	80,56%	69,94%
CAT, CS, Monotonia, CCT28	0,2	80,56%	66,32%
CAT, CS, Tensão, CCT28	0,2	80,56%	65,26%
CAT, Monotonia, Tensão, RAC28	0,3	80,56%	61,29%

Fonte: Autoria própria.

De acordo com os resultados da Etapa 1, detalhados na Tabela 3.15, foram observados altos valores de sensibilidade, acompanhados, no entanto, por baixa acurácia, como a primeira combinação da Tabela 3.15. Dessa forma, foram selecionadas as combinações que obtiveram sensibilidade maiores ou iguais a 75% e acurácia maiores ou iguais a 80% através do método DENFIS, apresentadas na Tabela 3.16.

Tabela 3.16: Combinações de variáveis com valores de sensibilidade maiores ou iguais a 75% e acurácia maiores ou iguais a 80% pelo método DENFIS.

Variáveis	Ponto de corte	Métricas	
		Sensibilidade	Acurácia
CD, Monotonia, Tensão, Duração do Sono	0,4	75,00%	85,38%
CD, CCT28, RAC28, Duração do Sono	0,3	75,00%	82,11%
CD, Monotonia, Tensão, Duração do Sono	0,3	75,00%	81,29%

Fonte: Autoria própria.

Dentre todas as combinações testadas, aquela que obteve os maiores valores de sensibilidade e acurácia foi o modelo com "Carga Diária", "Monotonia", "Tensão" e "Duração do Sono" como variáveis de entrada, como observado na Tabela 3.16. Além disso, as variáveis Monotonia, Tensão e Duração do Sono possuem fundamentações teóricas que demonstram seu potencial na predição de lesão, como visto na Tabela 3.2. É possível observar que o ponto de corte dessa combinação foi 0,4. Dessa forma, essa combinação com ponto de corte a 0,4 foi escolhida para ser testada na Etapa 2, a qual consiste na escolha dos hiperparâmetros que proporcionem o melhor desempenho ao modelo DENFIS. Nessa segunda etapa do processo, foram avaliadas todas as combinações dos hiperparâmetros do DENFIS apresentados anteriormente, totalizando 135 combinações. Na Tabela 3.17 são apresentadas as oito combinações de hiperparâmetros da Etapa 2 com ponto de corte 0,4 que apresentaram os maiores valores de sensibilidade.

Tabela 3.17: Oito combinações de hiperparâmetros com ponto de corte a 0,4 que apresentaram os maiores valores de sensibilidade pelo Método DENFIS.

Dthr	Máx. de iterações	Tamanho do passo	d	Corte	Métricas	
					Sensibilidade	Acurácia
0,050	300	0,02	2,0	0,4	77,78%	81,99%
0,050	200	0,02	2,0	0,4	77,78%	81,75%
0,050	100	0,02	2,0	0,4	77,78%	81,64%
0,050	300	0,01	2,0	0,4	77,78%	81,64%
0,050	200	0,01	2,0	0,4	77,78%	81,52%
0,050	300	0,03	3,0	0,4	75,00%	85,38%
0,050	100	0,03	2,5	0,4	75,00%	85,26%
0,050	200	0,03	2,5	0,4	75,00%	85,26%

Fonte: Autoria própria.

Observe que em todas as 8 melhores combinações, apresentadas na Tabela 3.17, aparecem unicamente o hiperparâmetro $Dthr = 0{,}05$, variando apenas os parâmetros "Máximo de iterações", "Tamanho do passo" e "d".

Cada uma dessas oito combinações de hiperparâmetros apresentadas na Tabela 3.17 foram então utilizadas para treinar o modelo com validação cruzada k-*fold* ($k = 5$) no conjunto de dados de treinamento, a fim de avaliar a generalização do modelo e evitar a ocorrência de *overfitting*. Na Tabela 3.18 são apresentadas as médias de sensibilidade e acurácia dos 5 *folds* obtidas pela validação cruzada, realizada nos modelos com as combinação de hiperparâmetros escolhidas na etapa anterior (Tabela 3.17).

Tabela 3.18: Validação Cruzada aplicada nas combinações da Tabela 3.17.

Dthr	Máx. iter.	Tam. passo	d	Médias das métricas da validação cruzada	
				Sensibilidade	Acurácia
0,05	300	0,03	3,0	78,28%	84,68%
0,05	200	0,03	2,5	77,17%	84,18%
0,05	100	0,03	2,5	74,95%	83,73%
0,05	300	0,02	2,0	74,67%	83,18%
0,05	200	0,02	2,0	74,67%	82,98%
0,05	300	0,01	2,0	74,67%	82,92%
0,05	200	0,01	2,0	74,67%	82,72%
0,05	100	0,02	2,0	74,67%	82,47%

Fonte: Autoria própria.

Com base nos resultados da validação cruzada, o modelo DENFIS com os hiperparâmetros $Dthr$ = 0,05, Máximo de iterações = 300, Tamanho do passo = 0,03 e d = 3 foi selecionado como o modelo final para predição do risco de lesão, por obter a maior média de sensibilidade e acurácia realizadas nos 5 *folds*.

Na Tabela 3.19 são apresentados os valores de sensibilidade e acurácia obtidos pelo modelo final DENFIS em cada um dos 5 *folds* da validação cruzada.

Tabela 3.19: Valores de sensibilidade e acurácia obtidos em cada *fold* da validação cruzada.

Fold	Métricas	
	Sensibilidade	Acurácia
1	88,89%	83,25%
2	77,78%	85,46%
3	69,23%	85,96%
4	73,68%	84,21%
5	81,82%	84,50%
Média	78,28%	84,68%
Desvio Padrão	±7,56%	±1,07%

Fonte: Autoria própria.

Na Tabela 3.20 é apresentado um resumo do modelo final DENFIS.

Tabela 3.20: Modelo final DENFIS para predição do risco de lesão.

Variáveis de entrada	Dthr	Máx. iter.	Tam. passo	d	Validação Cruzada	
					Sens.	Acur.
Carga Diária, Monotonia, Tensão, Duração do Sono	0,05	300	0,03	3	78,28%	84,68%

Fonte: Autoria própria.

Na Tabela 3.21 é representada a matriz de confusão do modelo de predição do risco de lesão gerado pelo DENFIS para os dados de teste, sendo as linhas os valores reais e as colunas os valores preditos.

Tabela 3.21: Matriz de confusão do DENFIS nos dados de teste.

	0	**1**
0	703	116
1	9	27

O modelo DENFIS classificou corretamente 703 atletas da classe sem lesão (0) e 27 com lesão (1), totalizando 730 classificações corretas, o que corresponde a uma acurácia de 85,38% no conjunto de teste. Além disso, o modelo identificou corretamente 27 das 36 atletas lesionadas, resultando em uma sensibilidade de 75%.

O sistema desenvolvido permite a avaliação individualizada de cada atleta, uma vez que é possível determinar o risco de lesão para jogadoras a partir de novos dados de entrada que estejam nas mesmas condições. Para ilustrar a aplicação do modelo, é possível analisar o caso de três atletas distintas. Na Tabela 3.22 são mostrados os dados de entrada para cada uma delas. A partir dessas informações, o SBRF gerado pelo DENFIS estimou o risco de lesão para cada uma, que são apresentados na Tabela 3.22.

Tabela 3.22: Variáveis de entrada e risco para cada atleta gerado pelo modelo DENFIS.

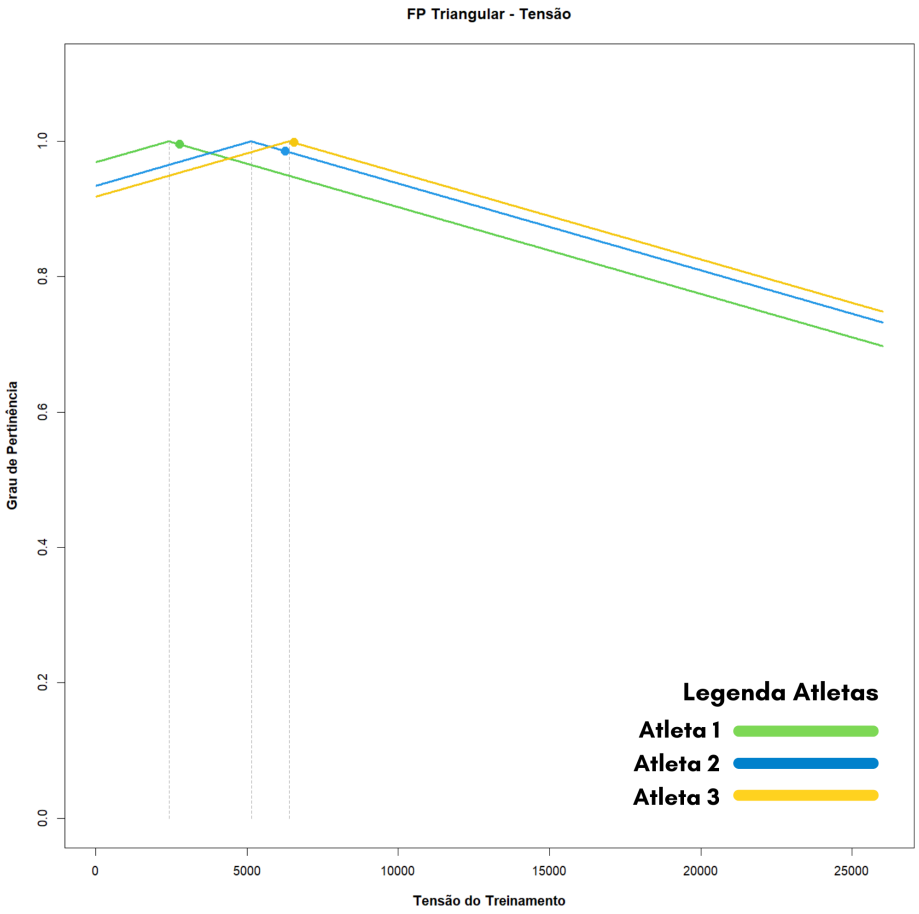
Caso	Variáveis de entrada				Variável de Saída
	CD	Monotonia	Tensão	Duração do sono (h)	Risco de Lesão (%)
Atleta 1	360	1,24	2790,01	8,0	0%
Atleta 2	300	1,63	6275,50	8,5	59,23%
Atleta 3	90	1,99	6561,03	7,0	98,88%

Fonte: Autoria própria.

A Atleta 1 apresenta baixos valores de monotonia e tensão, além de uma duração de sono adequada, o que resulta em um baixo risco de lesão predito pelo modelo DENFIS (0%). Por outro lado, a Atleta 2 possui valores moderados de monotonia e tensão, juntamente com uma boa duração de sono, resultando em um risco de lesão estimado em 59,23%, indicando um risco moderado de lesão. Por fim, a Atleta 3 apresenta baixo valor de carga diária, mas acompanhado de altos níveis de monotonia e tensão, além de uma duração de sono abaixo do ideal, o que contribuiu para a predição de um alto risco de lesão, igual a 98,88%.

Cada uma das atletas da Tabela 3.22 pertence a um *cluster* diferente, determinado pelo método ECM. Cada *cluster* representa uma regra fuzzy do sistema, associada a um nível de risco de lesão. E cada termo linguístico de cada variável da regra, por sua vez, é um conjunto fuzzy, representado por uma função de pertinência do tipo triangular. Por exemplo, na Figura 3.5 são apresentadas as funções de pertinência dos termos linguísticos da variável "Tensão de Treinamento" das regras geradas pelos *clusters* das atletas da Tabela 3.22.

Figura 3.5: Funções de pertinência dos termos linguísticos da variável "Tensão de treinamento" das regras geradas pelos *clusters* das atletas da Tabela 3.22.

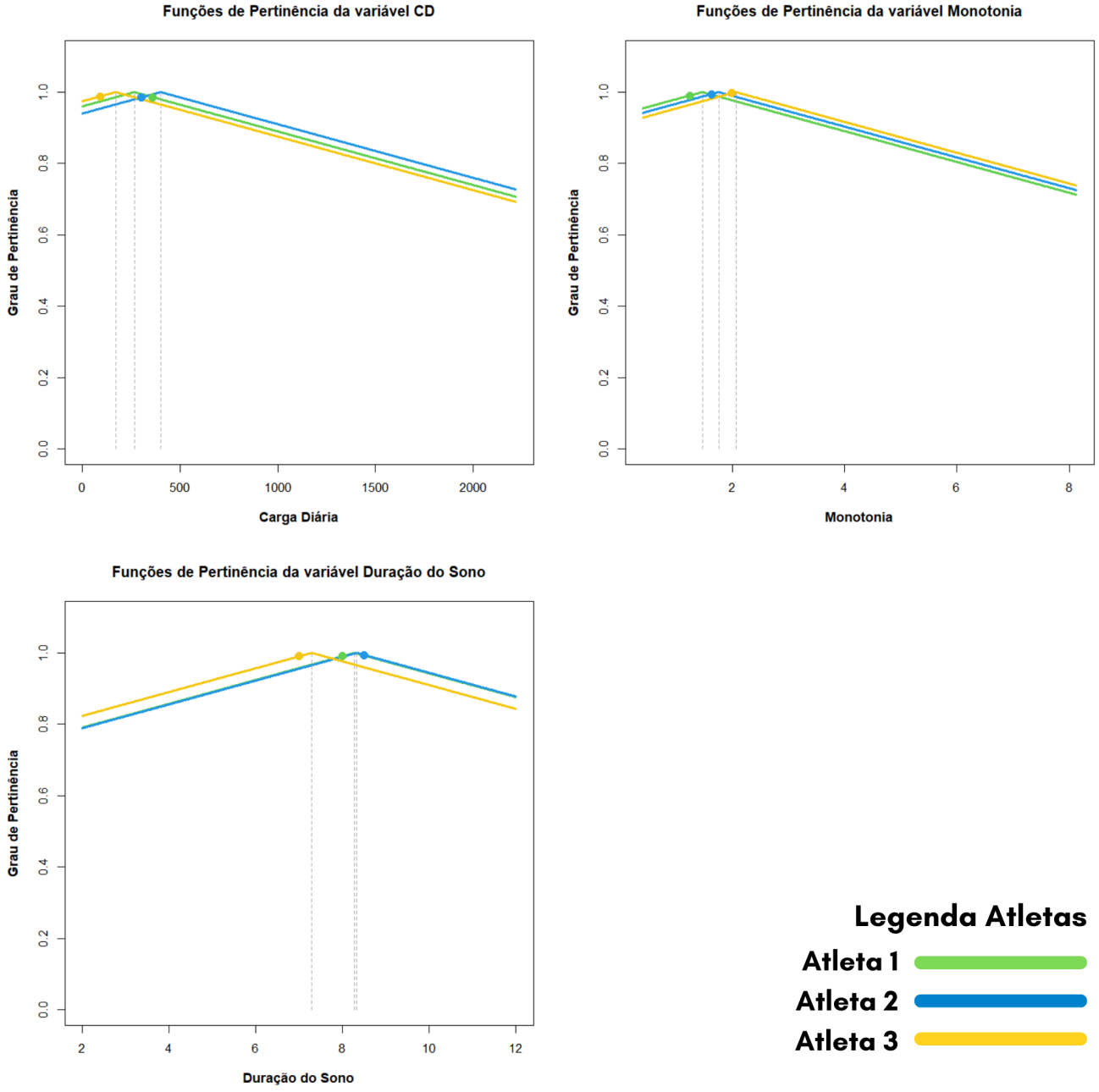


Fonte: Autoria própria.

A Atleta 1, representada pelo ponto verde na Figura 3.5, possui baixa tensão de treinamento (2790,01) e a função de pertinência assume o valor máximo no centro do *cluster*, representada pela função triangular ilustrada na cor verde na Figura 3.5. A Atleta 2, representada pelo ponto amarelo na Figura 3.5, apresenta uma tensão alta (6275,50) e a função de pertinência associada ao *cluster* é representada pela cor azul na Figura 3.5. Por fim, a Atleta 3, representada pelo ponto amarelo na Figura 3.5, possui uma tensão de treinamento muito alta (6561,03) e a função de pertinência associada ao *cluster* é representada pela cor amarela na Figura 3.5.

As funções de pertinência dos termos linguísticos das outras variáveis das regras geradas pelos *clusters* das atletas da Tabela 3.22 são apresentadas na Figura 3.6.

Figura 3.6: Funções de pertinência dos termos linguísticos das variáveis "Carga Diária", "Monotonia", "Duração do Sono" e "Risco de Lesão" das regras geradas pelos *clusters* das atletas da Tabela 3.22.



Fonte: Autoria própria.

Neste modelo são construídos 85 *clusters*, cada um com regra fuzzy associada. A Atleta 2 pertence ao *cluster* de número 3 gerado pelo DENFIS, de centro (401, 1,76, 5.141, 8,3), com os termos linguísticos de cada variável representados pelas funções de pertinência na Figura 3.5 e 3.6 da cor azul.

3.3.2 Resultados Obtidos pelo SBC/FCM

Nesta subseção são apresentados os resultados obtidos através do método SBC/FCM nas 3 Etapas apresentadas na Figura 3.3. A Tabela 3.23 apresenta as 15 combinações de variáveis de entrada com os maiores valores de sensibilidade e seus respectivos valores de acurácia obtidos pelo método SBC/FCM, resultados da Etapa 1 da análise.

Tabela 3.23: Desempenho das 15 combinações de variáveis com maiores valores de sensibilidade pelo método SBC/FCM.

Variáveis	Ponto de corte	Métricas	
		Sensibilidade	Acurácia
CAT, CS, CCT28, Duração do Sono	0,4	75,00%	91,81%
CAT, CS, CCT28, Duração do Sono	0,3	75,00%	90,99%
CAT, CS, CCT28, Duração do Sono	0,25	75,00%	90,41%
CAT, CS, CCT28, Duração do Sono	0,2	75,00%	89,82%
CAT, CS, CCT28, RAC28	0,4	75,00%	85,73%
CAT, CS, CCT28, RAC28	0,3	75,00%	83,39%
CAT, CS, CCT28, RAC28	0,25	75,00%	80,70%
CAT, CS, CCT28, RAC28	0,2	75,00%	78,13%
CAT, CS, CCT28, Duração do Sono	0,5	72,22%	91,81%
CAT, CS, CCT28, RAC28	0,5	72,22%	89,59%
CAT, CS, RAC42, RAC28	0,2	72,22%	83,98%
CAT, CS, Monotonia, Tensão	0,2	72,22%	65,61%
CAT, CS, CCT28, RAC42	0,5	69,44%	92,05%
CD, CAT, CS, Tensão	0,5	69,44%	91,93%
CD, CAT, CS, Monotonia	0,5	69,44%	91,35%

Fonte: Autoria própria.

Dentre todas as combinações testadas, aquela que obteve os maiores valores de sensibilidade e acurácia foi o modelo com "Carga Aguda de Treinamento", "Carga Semanal", "Carga Crônica de Treinamento 28" e "Duração do Sono" como variáveis de entrada, como observado na Tabela 3.23. Além disso, as variáveis CAT, CCT28 e Duração do Sono possuem fundamentações teóricas que demonstram seu potencial na predição de lesão, como visto na Tabela 3.2. É possível observar que o ponto de corte dessa combinação foi 0,4. Dessa forma, essa combinação com ponto de corte a 0,4 foi escolhida para ser testada na Etapa 2, a qual consiste na escolha dos hiperparâmetros que proporcionem o melhor desempenho ao modelo SBC/FCM. Foram avaliadas todas as combinações dos hiperparâmetros apresentados anteriormente, totalizando 24 combinações. Na Tabela 3.24 são apresentadas as oito combinações de hiperparâmetros da Etapa 2 com ponto de corte 0,4 que apresentaram os maiores valores de sensibilidade.

Tabela 3.24: Oito combinações de hiperparâmetros com ponto de corte a 0,4 que apresentaram os maiores valores de sensibilidade pelo Método SBC/FCM.

r.a.	eps.high	eps.low	Corte	Métricas	
				Sensibilidade	Acurácia
0,030	0,15	0,01	0,4	75,00%	91,81%
0,050	0,15	0,01	0,4	75,00%	91,81%
0,080	0,15	0,01	0,4	75,00%	91,81%
0,020	0,15	0,01	0,4	75,00%	91,78%
0,025	0,15	0,01	0,4	75,00%	91,70%
0,035	0,15	0,01	0,4	75,00%	91,70%
0,048	0,15	0,01	0,4	75,00%	91,70%
0,055	0,15	0,01	0,4	75,00%	91,70%

Fonte: Autoria própria.

Cada uma dessas oito combinações de hiperparâmetros apresentadas na Tabela 3.24 foram então utilizadas para treinar o modelo com validação cruzada k-*fold* ($k = 5$) no conjunto de dados de treinamento, a fim de avaliar a generalização do modelo e evitar a ocorrência de *overfitting*. Na Tabela 3.25 são apresentadas as médias de sensibilidade e acurácia dos 5 *folds* obtidas pela validação cruzada, realizada nos modelos com as combinação de hiperparâmetros escolhidas na etapa anterior (Tabela 3.24).

Tabela 3.25: Média das métricas da validação cruzada aplicada nas combinações da Tabela 3.24.

r.a.	eps. high	eps.low	Métricas da validação cruzada	
			Sensibilidade	Acurácia
0,035	0,15	0,01	61,56%	90,77%
0,030	0,15	0,01	61,56%	90,67%
0,025	0,15	0,01	61,56%	90,66%
0,020	0,15	0,01	61,56%	90,63%
0,080	0,15	0,01	61,56%	89,63%
0,050	0,15	0,01	60,79%	90,78%
0,048	0,15	0,01	60,79%	90,78%
0,055	0,15	0,01	60,79%	90,74%

Fonte: Autoria própria.

Com base nos resultados da validação cruzada, o modelo SBC/FCM com os hiperparâmetros $r.a. = 0{,}035$, $eps.high = 0{,}15$ e $eps.low = 0{,}01$ foi selecionado como o modelo final para predição do risco de lesão por obter a maior média de sensibilidade e acurácia realizada nos 5 *folds*. Na Tabela 3.26 são apresentados os valores de sensibilidade e acurácia obtidos pelo modelo final SBC/FCM em cada um dos 5 *folds* da validação cruzada.

Tabela 3.26: Valores de sensibilidade e acurácia obtidos em cada *fold* da validação cruzada.

Fold	Métricas	
	Sensibilidade	Acurácia
1	55,56%	91,75%
2	61,11%	88,44%
3	50,00%	91,94%
4	68,42%	89,47%
5	72,73%	92,23%
Média	61,56%	90,77%
Desvio Padrão	±9,24%	±1,70%

Fonte: Autoria própria.

Na Tabela 3.27 é apresentado um resumo do modelo final SBC/FCM.

Tabela 3.27: Modelo final SBC/FCM para predição do risco de lesão.

Variáveis de entrada	r.a.	eps. high	eps.low	Validação Cruzada	
				Sens.	Acur.
Carga Semanal, Carga Aguda, Carga Crônica 28, Duração do Sono	0,035	0,15	0,01	61,56%	90,77%

Fonte: Autoria própria.

Na Tabela 3.28 é apresentada a matriz de confusão do modelo de predição do risco de lesão gerado pelo SBC/FCM para os dados de teste.

Tabela 3.28: Matriz de confusão do SBC/FCM nos dados de teste.

	0	**1**
0	757	62
1	9	27

O modelo SBC/FCM classificou corretamente 757 atletas da classe sem lesão (0) e 27 com lesão (1), totalizando 784 classificações corretas, o que corresponde a uma acurácia de 91,70% no conjunto de teste. Além disso, assim como no DENFIS, o modelo gerado pelo SBC/FCM identificou corretamente 27 das 36 atletas lesionadas, resultando em uma sensibilidade de 75%.

O sistema desenvolvido também permite a avaliação individualizada de cada atleta, uma vez que é possível determinar o risco de lesão para jogadoras a partir de novos dados de entrada que estejam nas mesmas condições. Para ilustrar a aplicação do modelo, é possível analisar o caso das três atletas distintas, apresentadas na Tabela 3.29, e seus riscos foram estimados pelo modelo SBC/FCM. Na Tabela 3.29 são mostrados os dados de entrada para cada uma delas. A partir dessas informações, o SBRF gerado pelo SBC/FCM estimou o risco de lesão para cada uma, que também são apresentados na Tabela 3.29.

Tabela 3.29: Variáveis de entrada e risco para cada atleta gerado pelo modelo SBC/FCM.

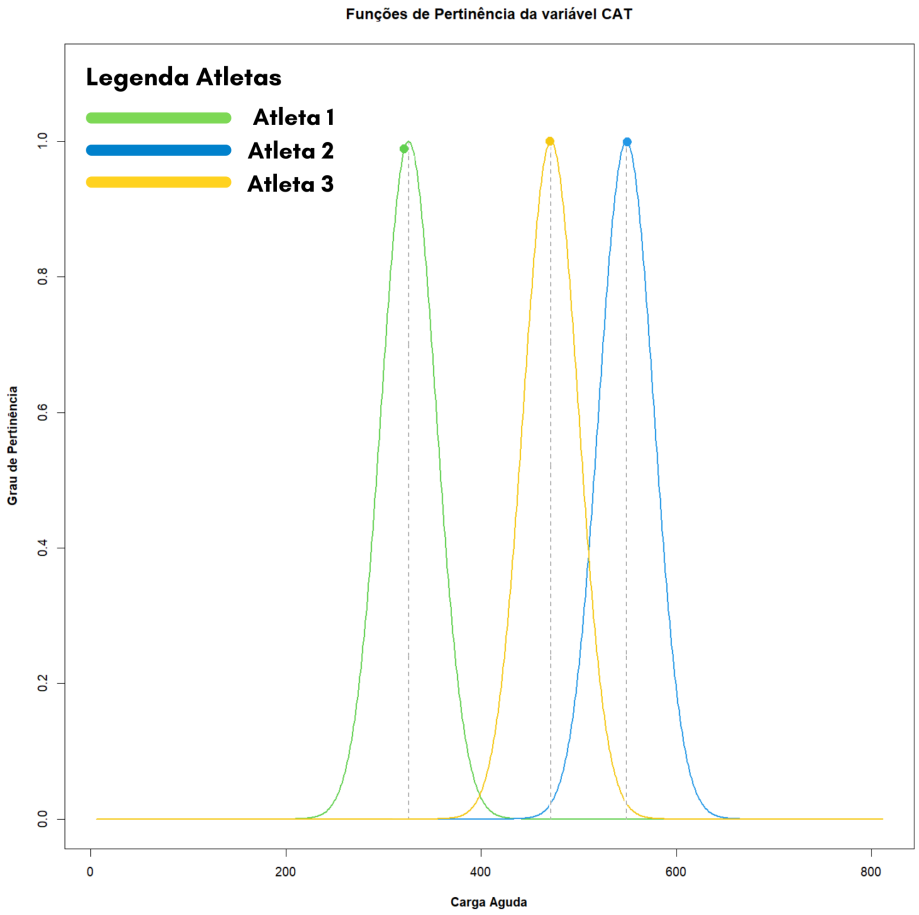
Caso	Variáveis de entrada				Variável de Saída
	CS	CAT	CCT28	Duração do sono (h)	Risco de Lesão (%)
Atleta 1	2250	321,43	319,29	8	5,7%
Atleta 2	3850	550	350	8,5	99,98%
Atleta 3	3297	471	500	7,0	99,99%

Fonte: Autoria própria.

A Atleta 1 apresenta valores de Carga Aguda e Carga Crônica próximos, indicando uma Relação Carga Aguda-Crônica equilibrada (RAC = 321,43/319,29 = 1,006) além de uma duração do sono considerada adequada, o que resulta em um baixo risco de lesão predito pelo modelo SBC/FCM (5,7%). Por outro lado, a Atleta 2 possui uma Carga Aguda significativamente maior que a Carga Crônica, obtendo uma Relação Carga Aguda-Crônica igual a 1,57, valor considerado de risco por estar fora da zona considerada segura por especialistas (entre 0,8 e 1,3) (WINDT; GABBETT, 2017). Isso indica um aumento abrupto na carga de treinamento, o que, mesmo acompanhado de valores adequados na duração do sono, resultou em um risco de lesão estimado em 99,98%, indicando um risco elevado de lesão. Por fim, a Atleta 3 apresenta valores de Carga Aguda e Carga Crônica próximos, indicando uma Relação Carga Aguda-Crônica equilibrada (RAC = 471/500 = 0,942), porém com um alto valor carga semanal e uma duração do sono abaixo do ideal, o que contribuiu para a predição de um alto risco de lesão, igual a 99,99%.

Cada uma das atletas da Tabela 3.29 pertence a um *cluster* diferente, determinado pelo método SBC. Cada *cluster* representa uma regra fuzzy do sistema, associada a um nível de risco de lesão. E cada termo linguístico de cada variável da regra, por sua vez, é um conjunto fuzzy, representado por uma função de pertinência do tipo gaussiana. Por exemplo, na Figura 3.7 são apresentadas as funções de pertinência dos termos linguísticos da variável "Carga Aguda de Treinamento" das regras geradas pelos *clusters* das atletas da Tabela 3.29.

Figura 3.7: Funções de pertinência dos termos linguísticos da variável "Carga Aguda de Treinamento" das regras geradas pelos *clusters* das atletas da Tabela 3.29.



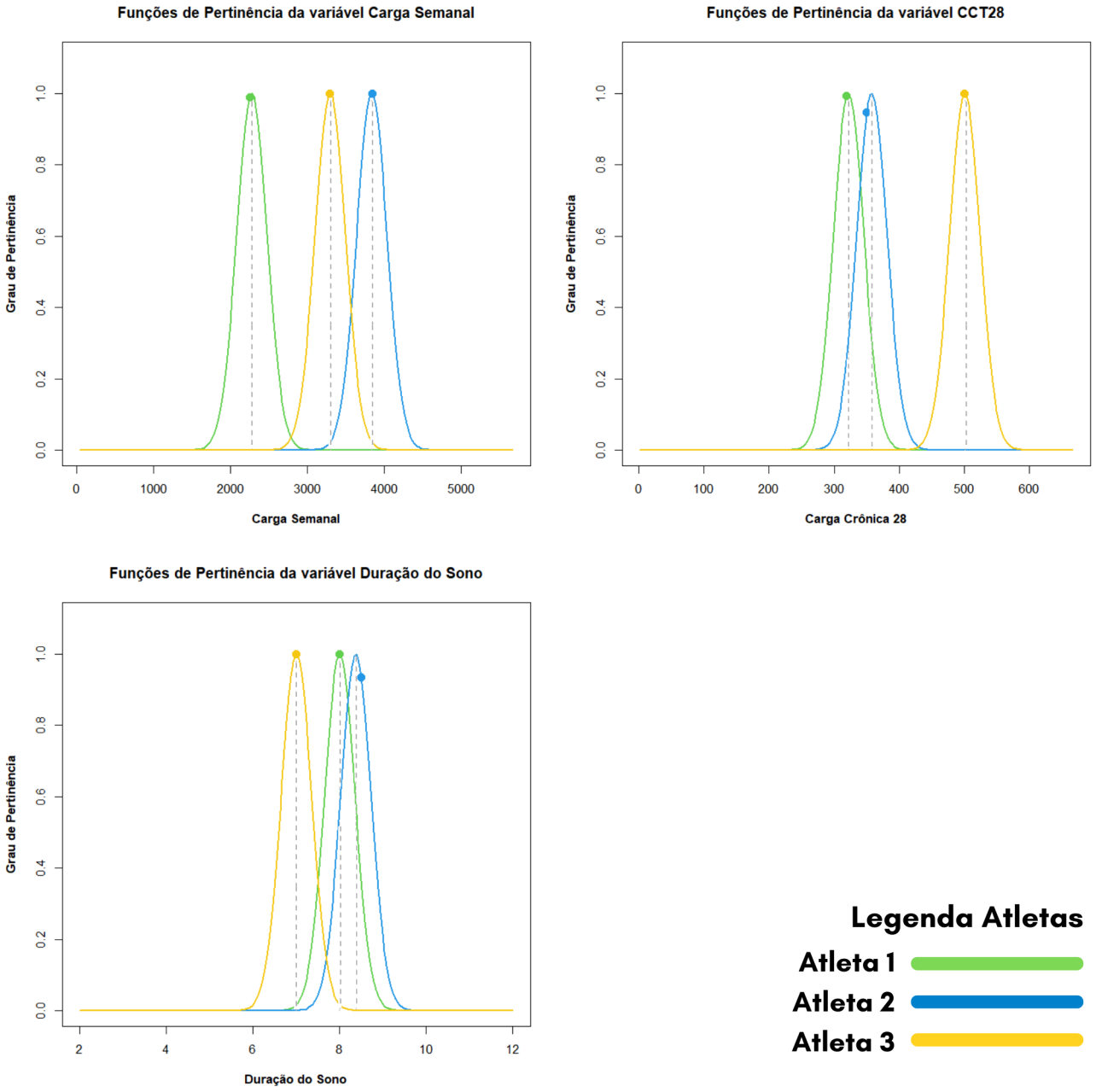
Fonte: Autoria própria.

A Atleta 1, representada pelo ponto verde na Figura 3.7, possui baixa tensão de treinamento (2790,01) e a função de pertinência assume o valor máximo no centro do *cluster* e é representada

pela função triangular ilustrada na cor verde na Figura 3.7. A Atleta 2, representada pelo ponto amarelo na Figura 3.7, apresenta uma tensão alta (6275,50) e a função de pertinência associada ao *cluster* é representada pela cor azul na Figura 3.7. Por fim, a Atleta 3, representada pelo ponto amarelo na Figura 3.7, possui uma tensão de treinamento muito alta (6561,03) e a função de pertinência associada ao *cluster* é representada pela função de pertinência da cor amarelo na Figura 3.7.

As funções de pertinência dos termos linguísticos das outras variáveis das regras geradas pelos *clusters* das atletas da Tabela 3.29 são apresentadas na Figura 3.8.

Figura 3.8: Funções de pertinência dos termos linguísticos das variáveis "Carga Semanal", "Carga Crônica de Treinamento 28" e "Duração do Sono" das regras geradas pelos *clusters* das atletas da Tabela 3.29.



Fonte: Autoria própria.

Neste modelo são construídos 2154 *clusters*, cada um com regra fuzzy associada. A Atleta

2 pertence ao *cluster* de número 3 gerado pelo SBC/FCM, de centro $(3841, 549, 358, 8{,}4)$, com os termos linguísticos de cada variável representados pelas funções de pertinência apresentadas na Figura 3.7 e 3.8 da cor azul.

3.3.3 Comparação entre os Métodos DENFIS e SBC/FCM

Nesta subseção é realizada a comparação entre os dois métodos de redes neuro-fuzzy desenvolvidos neste trabalho: DENFIS e SBC/FCM.

- Etapa 1:
  - Os modelos escolhidos nessa etapa pelo método DENFIS e SBC/FCM alcançaram mesma sensibilidade de 75%, porém o modelo do SBC/FCM obteve acurácia maior de 91,81% em comparação a 85,38% do DENFIS.
  - O método SBC/FCM obteve 7 combinações de variáveis e ponto de corte que alcançaram sensibilidade maior ou igual a 75% e acurácia maior ou igual a 80%.
  - O método DENFIS obteve apenas 3 combinações com esse desempenho, conforme apresentado nas Tabelas 3.16 e 3.23.

  Os modelos utilizaram as seguintes variáveis:

  - DENFIS: "Carga Diária", "Monotonia", "Relação Carga Aguda-Crônica" e "Duração do Sono".
  - SBC/FCM: "Carga Semanal", "Carga Aguda de Treinamento", "Carga Crônica de Treinamento 28" e "Duração do Sono".

  Ambas as combinações possuem fundamentações teóricas que demonstram seu potencial na predição de lesão, como visto na Tabela 3.2. Além disso, os dois modelos utilizam a variável "Duração do Sono", o que reforça a importância dessa variável na predição do risco de lesão.

  Em relação ao ponto de corte na Etapa 1:

  - Ambos os métodos indicaram o ponto de corte 0,4 como aquele que proporcionou o melhor desempenho (entre os modelos com sensibilidade maior ou igual a 75% e acurácia maior ou igual a 80%) na predição do risco de lesão, conforme apresentado nas Tabelas 3.16 e 3.23.

  Outra observação importante nas Tabelas 3.15 e 3.23 é a quantidade de cada ponto de corte presente entre as 15 combinações das tabelas. Note que:

  - No método SBC/FCM, os pontos de corte 0,5 e 0,4 está presente em 7 das 15 combinações (Tabela 3.23).
  - No método DENFIS, esses pontos de cortes não aparecem entre as 15 combinações com maiores sensibilidades (ver Tabela 3.15)

  Tal fato pode sugerir que o método DENFIS é mais sensível ao diminuir o ponto de corte em comparação com o método SBC/FCM, podendo indicar que o método SBC/FCM faz predições mais conservadoras, ou seja, prevendo poucos atletas com riscos intermediários (valores entre 0,4 e 0,5) e mais atletas com riscos altos (próximos a 1) e baixo (próximos a 0) em comparação ao DENFIS.

Destaca-se que, embora a redução do ponto de corte possa aumentar a sensibilidade, também tende a aumentar a quantidade de falsos positivos, ou seja, atletas sem risco de lesão (classe 0) sendo classificados como alto risco de lesão (classe 1). A escolha de considerar outros valores como ponto de corte de arredondamento para classe 1, tem como objetivo adotar uma abordagem preventiva ao modelo, priorizando indicar atletas com alto risco de lesão, porém sem aumentar excessivamente a quantidade de falsos positivos, o que poderia ter como consequência o controle de carga e o resguardo de atletas em partidas de maneira equivocada e sem necessidade. Dessa forma, é importante escolher um ponto de corte que mantenha um equilíbrio entre as métricas sensibilidade e acurácia.

- Etapa 2:

  A segunda etapa tem por objetivo encontrar a combinação de hiperparâmetros que proporcionasse o melhor desempenho nas métricas sensibilidade e acurácia, fixando o ponto de corte a 0,4 como indicado na Etapa 1. Nas 8 melhores combinações de hiperparâmetros apresentadas nas Tabelas 3.17 e 3.24, é possível observar os seguintes pontos:

  - O método DENFIS conseguiu em algumas combinações de hiperparâmetros aumentar sua sensibilidade para 77,78%, entretanto sua acurácia foi reduzida para valores próximos a 81%.
  - O método SBC/FCM manteve a sensibilidade em 75% e sua acurácia em 91,81%.

- Etapa 3: Na Etapa 3, as oito combinações de hiperparâmetros selecionadas na Etapa 2, apresentadas nas Tabelas 3.17 e 3.24, foram submetidas à validação cruzada k-*fold* ($k = 5$) para avaliar a generalização dos modelos. As métricas finais obtidas pela validação cruzada, apresentadas nas Tabelas 3.20 e 3.27, sugerem os seguintes pontos:

  - O modelo DENFIS, quanto ao critério sensibilidade e desvio padrão nas métricas dos *folds*, manteve seu desempenho superior, apresentando sensibilidade de 78,28% e acurácia de 84,68%, além de um desvio padrão de 7,56% na sensibilidade e 1,07% na acurácia dos *folds*.
  - O modelo SBC/FCM alcançou sensibilidade de 61,56% e acurácia de 90,77%, além de um desvio padrão de 9,24% na sensibilidade e 1,70% na acurácia dos *folds*.

  A Tabela 3.30 apresenta os resultados da validação cruzada dos modelos finais de cada método, escolhidos após a análise detalhada anteriormente.

Tabela 3.30: Comparação do modelo final DENFIS com o modelo final SBC/FCM para predição do risco de lesão.

Método	Sensibilidade	Acurácia
DENFIS	78,28%	84,68%
SBC/FCM	61,56%	90,77%

Fonte: Autoria própria.

De acordo com os resultados apresentados na Tabela 3.30 e considerando que o objetivo principal deste trabalho é a predição do risco de lesões em atletas de futebol feminino, o modelo obtido pelo DENFIS pode ser o mais adequado neste caso para ser implementado em um sistema de monitoramento de atletas. Tal afirmação fundamenta-se no desempenho superior deste modelo, obtido na métrica de sensibilidade, resultando na diminuição da ocorrência de falsos

negativos, ou seja, atletas com risco real de lesão (1) indevidamente classificados como de baixo risco (0). Essa característica é de extrema importância em modelos de prevenção de lesões, pois um falso negativo pode indicar uma lesão iminente que poderia não ter sido previamente detectada, de modo que a atleta continue sendo submetida a cargas de treinamento elevadas causando danos à sua saúde e à sua carreira.

3.3.4 Comparação dos Riscos de Novas Atletas

Para auxiliar na comparação entre os modelos e ter valores como referência, são apresentados na Tabela 3.31 os valores mínimos, máximos e a média de cada variável de entrada utilizada nos modelos finais de predição de lesão.

Tabela 3.31: Valores mínimos, máximos e média de cada variável de entrada utilizada nos modelos finais de predição de lesão.

	Variáveis de entrada						
	CD	CS	CAT	CCT28	Monotonia	Tensão	Duração do sono
Mínimo	0	40	5,71	1,43	0,410	16,4	2
Média	507,6	2348	335,44	314,98	1,262	3320,4	8,098
Máximo	2220	5680	811,43	666,79	8,130	26016	12

Fonte: Autoria própria.

Na Tabela 3.32 é apresentada uma comparação entre o risco de lesão predito pelos modelos DENFIS e SBC/FCM para três atletas distintas já apresentadas anteriormente.

Tabela 3.32: Comparação do risco de lesão de novas atletas predito pelos modelos DENFIS e SBC/FCM.

Atleta	Variáveis de entrada							Risco de Lesão (%)	
	CD	CS	CAT	CCT28	Monotonia	Tensão	DS	DENFIS	SBC/FCM
1	360	2250	321,43	319,29	1,24	2790,01	8,0	0%	5,7%
2	300	3850	550,00	350,00	1,63	6275,50	8,5	59,23%	99,98%
3	90	3297	471,00	500,00	1,99	6561,03	7,0	98,88%	99,99%

Fonte: Autoria própria.

A partir dos riscos preditos pelos modelos DENFIS e SBC/FCM apresentados na Tabela 3.32, é possível observar as características de cada modelo:

- Nota-se que na Atleta 1, que apresenta cargas de treinamento equilibradas e uma boa duração do sono, ambos os modelos calcularam um risco de lesão muito baixo, próximo a 0%.

- A Atleta 2, que possui uma carga aguda significativamente maior que a crônica, justifica a predição de um risco elevado pela SBC/FCM (99,98%), o que não ocorreu com o modelo DENFIS, que previu um risco moderado (59,23%).

  Essa diferença pode ser explicada pelo fato de o SBC/FCM fazer predições mais extremas, como observado anteriormente na discussão sobre a sensibilidade do método em relação ao ponto de corte (ver Tabela 3.23), enquanto o DENFIS tende a apresentar predições mais graduais. Apesar dessa diferença, ambos predizeram um risco considerável de lesão para a Atleta 2.

- Por fim, a Atleta 3 teve um risco de lesão elevado em ambos os modelos (98,88% pelo DENFIS e 99,1% pelo SBC/FCM).

  Mesmo não possuindo uma grande diferença entre a carga aguda e crônica, a predição do risco alto feita pela SBC/FCM pode ser justificada pela alta carga semanal e baixa duração do sono. O modelo DENFIS também previu um risco elevado para a Atleta 3, possivelmente devido à baixa duração do sono e alta monotonia e tensão de treinamento.

Na Tabela 3.33 é apresentado o risco de lesão predito pelos modelos DENFIS e SBC/FCM para uma nova atleta com baixa duração do sono.

Tabela 3.33: Variáveis de entrada e risco para cada atleta gerado pelo modelo DENFIS e SBC/FCM.

Atleta	Variáveis de entrada							Risco de Lesão (%)	
	CD	CS	CAT	CCT28	MON	Tensão	DS	DENFIS	SBC/FCM
A4	200	3990	570	280	2,5	9975	4,5	45,54%	0,00000009%
$\hat{A}4$	200	3990	570	280	2,5	9975	3,0	0%	0%

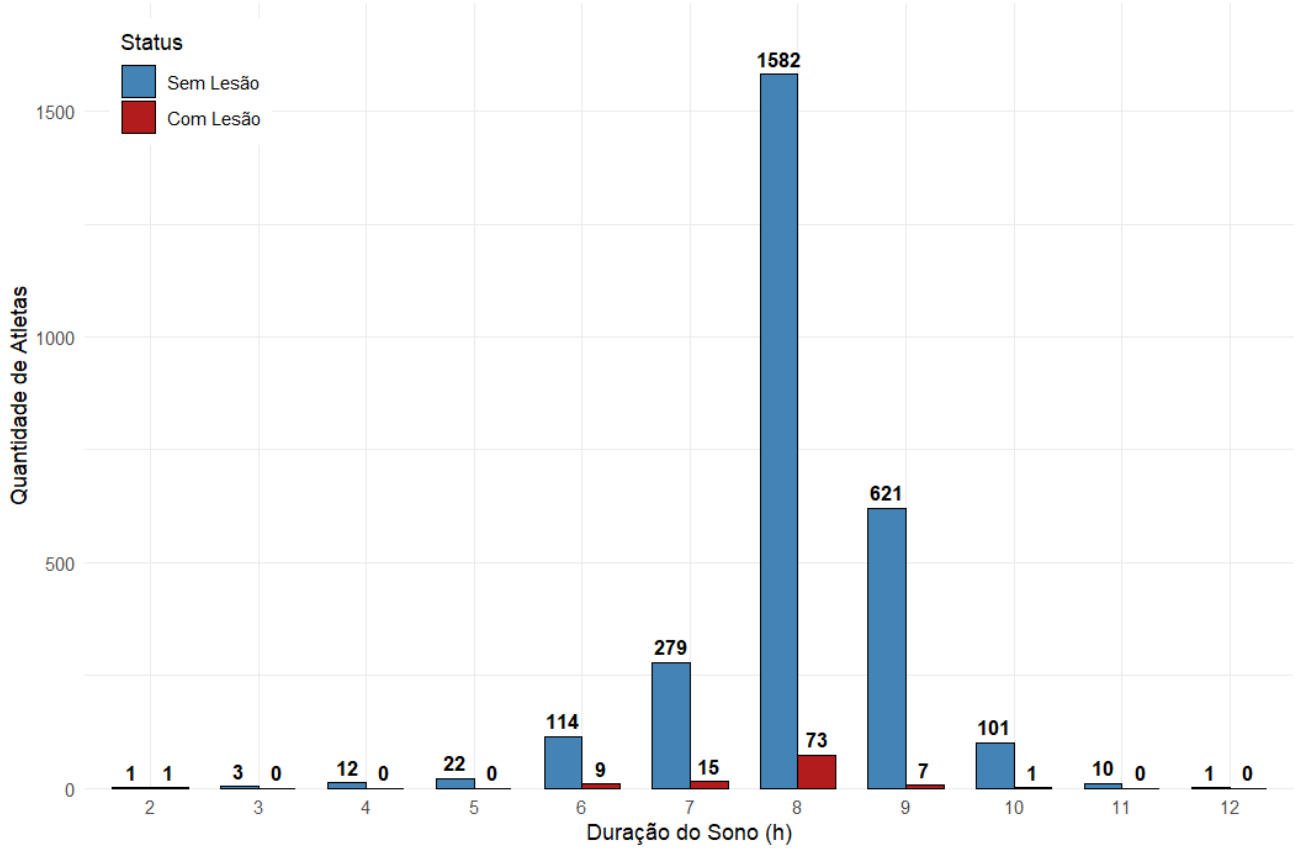
Fonte: Autoria própria.

Na Tabela 3.33, a Atleta 4 (A4) apresenta uma baixa duração do sono (4,5 horas) e uma alta monotonia, o que contribuiu para a predição de um risco moderado (45,54%) pelo modelo DENFIS. Por outro lado, mesmo com uma carga aguda significativamente maior que a crônica, o modelo SBC/FCM previu um risco extremamente baixo (0,00000009%). Tal fato pode sugerir que o modelo SBC/FCM não é expressivamente influenciado por baixos valores na duração do sono, ou seja, o modelo contradiz a literatura, uma vez que a baixa duração do sono é um fator conhecido por aumentar o risco de lesão em atletas (HARALDSDOTTIR *et al.*, 2021).

Além disso, para ilustrar a influência da duração do sono na predição do risco de lesão pelos modelos DENFIS e SBC/FCM, foi reduzida a duração de sono da Atleta 4 de 4,5 horas para 3 ($\hat{A}4$) mantendo os valores das outras variáveis. Nesse caso, ambos os modelos previram um risco de lesão ainda menor 0%, não realizando a predição de forma coerente com a literatura. Tal fato pode ter sido influenciado pela baixa quantidade de atletas com duração do sono igual ou inferior a 5 horas no conjunto de dados utilizado para o treinamento dos modelos, o que pode ter prejudicado a capacidade de generalização e aprendizado dos modelos para casos específicos como este, uma vez que os modelos aprendem com o banco de dados fornecido.

O gráfico da Figura 3.9 ilustra o número de registros de atletas não lesionadas e lesionadas para cada quantidade de horas na duração do sono registrada no conjunto de dados.

Figura 3.9: Relação entre a duração do sono e a ocorrência de lesão.



Fonte: Autoria própria.

É possível observar na Figura 3.9 que a maioria dos registros de atletas lesionadas possuem duração do sono entre 6 e 9 horas, sem nenhum registro entre 3 e 5 horas, e apenas uma atleta lesionada com 2 horas de sono. Portanto, é possível que os modelos DENFIS e SBC/FCM possam não ter aprendido adequadamente a relação entre baixa duração do sono (valores entre 2 e 5 horas) e risco de lesão, o que pode ter influenciado as predições da atleta $\hat{A}4$.

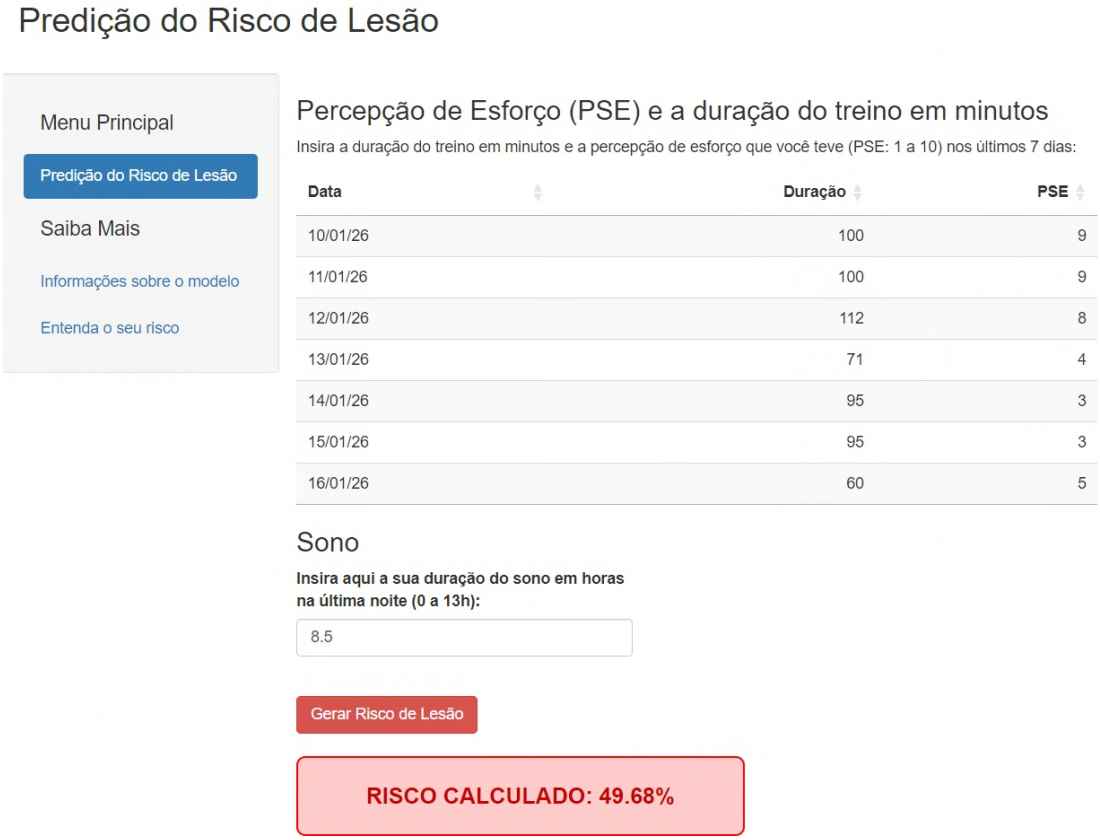
Na seção a seguir é apresentada a interface computacional desenvolvida para aplicação prática do modelo final DENFIS na predição do risco de lesão.

3.3.5 Interface Computacional

Foi desenvolvido um aplicativo através do pacote `Shiny` do *Software* R (CHANG *et al.*, 2025) que implementa o modelo final DENFIS para predição do risco de lesão em atletas de futebol feminino. Para se tornar intuitivo e de fácil utilização para os usuários, o aplicativo permite a entrada dos dados de treinamento, sendo estes a duração do treino (em minutos) e a percepção subjetiva de esforço (de 1 a 10) relativa ao treino realizado pela atleta nos últimos 7 dias. Além disso, o aplicativo também solicita a duração do sono (em horas) da atleta na noite anterior. A partir desses dados, o aplicativo calcula as variáveis necessárias para o modelo (Carga Diária, Monotonia e Tensão) e, com a duração do sono, utiliza o modelo final DENFIS (Tabela 3.20) para prever o risco de lesão da atleta.

A Figura 3.10 apresenta a interface do aplicativo desenvolvido. Para uma atleta com os valores de treino semanal e duração do sono apresentados na Figura 3.10, o modelo DENFIS previu um risco de lesão igual a 49,68%.

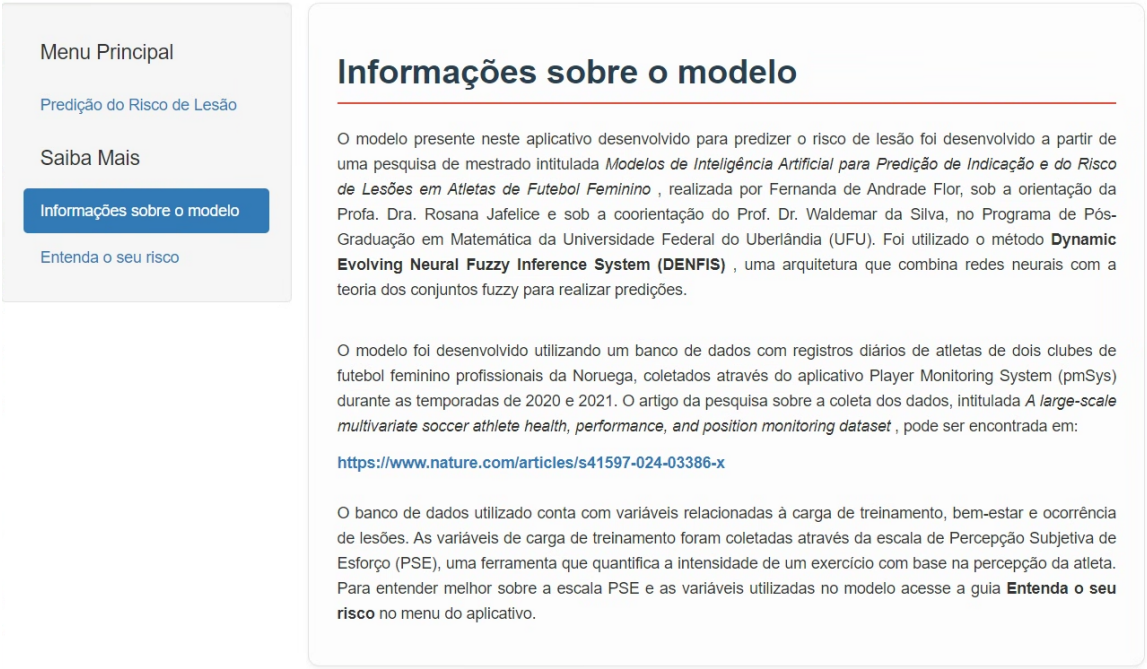
Figura 3.10: Página principal da interface computacional: predição do risco de lesão.



Fonte: Autoria própria.

Além da predição do risco, o aplicativo conta com uma seção "Saiba Mais", onde são apresentadas as subseções "Informações sobre o modelo" e "Entenda o seu risco", nas quais as interfaces são mostradas nas Figuras 3.11 e 3.12, respectivamente.

Figura 3.11: Subseção "Informações sobre o modelo" da interface computacional.



Fonte: Autoria própria.

Figura 3.12: Subseção “Entenda o seu risco” da interface computacional.

Fonte: Autoria própria.

O aplicativo foi desenvolvido com o objetivo de facilitar a aplicação prática do modelo desenvolvido neste trabalho, permitindo que atletas e treinadores possam monitorar o risco de lesão de maneira simples e eficiente. Atualmente é estudada a possibilidade de disponibilizar o aplicativo de maneira on-line.

No capítulo a seguir, são apresentadas as considerações finais deste trabalho.

Capítulo 4

Considerações Finais

Diante da natureza multifatorial das lesões esportivas, os avanços nas técnicas de aprendizado de máquina e inteligência artificial demonstram um elevado potencial na modelagem preditiva. Este trabalho apresenta originalidades, pois utiliza variáveis de carga de treinamento e bem-estar que são coletadas por especialistas da área da fisioterapia esportiva e, por meio de ferramentas computacionais, possibilita prever o risco de lesão em atletas, a qual é considerada uma variável de natureza incerta. A integração das redes neurais artificiais com teorias matemáticas, como os conjuntos fuzzy, resulta em sistemas com alta capacidade de modelar interações complexas. Nesse contexto, este trabalho adotou duas abordagens, que configuram o objetivo principal deste trabalho na construção de modelos preditivos sobre lesões esportivas em atletas de futebol feminino:

- Modelos de predição da indicação de lesão

  Neste trabalho, foram desenvolvidos modelos de predição de indicação de lesão esportiva (saída binária 0 ou 1) utilizando os métodos de aprendizado de máquina Árvore de Decisão e Floresta Aleatória. O método Floresta Aleatória, em concordância com a literatura, apresentou desempenho superior na predição de lesão, alcançando sensibilidade de 64,52% e acurácia de 98,71%. Os resultados de sensibilidade e acurácia sugerem uma melhor predição da classe "não lesão" (0) em comparação a classe "lesão" (1), o que pode ser explicado pela desproporção entre as classes no conjunto de dados utilizado para o treinamento dos modelos.

- Modelos de predição do risco de lesão

  Neste trabalho, foram desenvolvidos modelos de predição do risco de lesão esportiva (saída entre 0 e 1) utilizando o DENFIS e o SBC/FCM. Cada um desses métodos gera um SBRF para a predição do risco de lesão. O método DENFIS apresentou desempenho superior na predição do risco de lesão, alcançando sensibilidade de 78,28% e acurácia de 84,68% na validação cruzada. Os resultados obtidos pela sensibilidade e acurácia sugerem o potencial do modelo na predição do risco de lesão, principalmente na identificação de atletas com risco real de lesão.

O presente estudo contribui para a indicação das variáveis de carga de treinamento e bem-estar mais relevantes para a predição do risco de lesão em atletas de futebol feminino. Os modelos desenvolvidos através dos métodos DENFIS e SBC/FCM indicam as variáveis Carga Diária, Carga Semanal, Carga Aguda, Carga Crônica 28, Monotonia, Tensão e Duração do Sono como as mais relevantes para a predição do risco de lesão. Destaca-se neste trabalho a importância da variável Duração do Sono, indicada por ambos os modelos, reforçando o potencial dessa variável na predição do risco de lesão, conforme evidenciado na literatura.

Cabe ressaltar que, devido à natureza dos dados coletados, os modelos neuro-fuzzy desenvolvidos neste trabalho possuem limitações na predição do risco de lesão em atletas com baixa duração do sono (ver Figura 3.9), uma vez que o conjunto de dados utilizado para o treinamento dos modelos não possuía registros de atletas lesionadas com essa faixa de duração do sono, o que pode ter prejudicado a capacidade de generalização dos modelos, visto que estes aprendem com os dados fornecidos. Além disso, a precisão dos modelos desenvolvidos neste trabalho pode ser comprometida ao avaliar atletas que possuem variáveis com valores fora dos intervalos observados no conjunto de dados (ver Tabela 3.31). Portanto, sugere-se que trabalhos futuros possam explorar bancos de dados mais abrangentes, buscando melhorar a capacidade preditiva dos modelos.

Como trabalhos futuros, podem ser desenvolvidos modelos capazes de predizer, além do risco, a gravidade das lesões. Também, podem ser estudados dados de outras modalidades esportivas, que podem contribuir para a prevenção física de atletas e para a redução de gastos financeiros das entidades esportivas.

Este trabalho gerou duas publicações em anais de eventos, sendo uma no I Workshop de Matemática da UFU (FLOR; JAFELICE; SILVA, 2025) e outra na XIV Semana da Matemática da UFTM (FLOR *et al.*, 2025b), além de uma publicação de um pôster no 35º Colóquio Brasileiro de Matemática do IMPA (FLOR *et al.*, 2025a). Também será submetido um artigo para uma revista internacional sobre os modelos neuro-fuzzy na modelagem do risco de lesões esportivas em atletas de futebol feminino.

Referências Bibliográficas

BANDO, F. M. *Sistemas fuzzy e aproximação universal*. Dissertação (Mestrado) — IMECC, Unicamp, 2002.

BEDE, B. *Mathematics of Fuzzy Sets and Fuzzy Logic*. Berlin: Springer, 2013. v. 295. ISBN 978-3-642-35220-1. Disponível em: ⟨https://doi.org/10.1007/978-3-642-35221-8⟩.

BEZDEK, J. C.; EHRLICH, R.; FULL, W. Fcm: The fuzzy c-means clustering algorithm. *Computers & Geosciences*, v. 10, n. 2, p. 191–203, 1984. ISSN 0098-3004. Disponível em: ⟨https://doi.org/10.1016/0098-3004(84)90020-7⟩.

BISHOP, C. M. *Pattern Recognition and Machine Learning*. New York: Springer, 2006. ISBN 978-0387310732.

BITTENCOURT, N.; MEEUWISSE, W.; MENDONÇA, L.; NETTEL-AGUIRRE, A.; OCARINO, J.; FONSECA, S. Complex systems approach for sports injuries: moving from risk factor identification to injury pattern recognition—narrative review and new concept. *British Journal of Sports Medicine*, BMJ Publishing Group Ltd, v. 50, n. 21, p. 1309–1314, 2016. Disponível em: ⟨https://doi.org/10.1136/bjsports-2015-095850⟩.

BORG, G. *Borg's perceived exertion and pain scales*. Champaign, IL: Human Kinetics, 1998.

BOWYER, K. W.; CHAWLA, N. V.; HALL, L. O.; KEGELMEYER, W. P. SMOTE: synthetic minority over-sampling technique. *CoRR*, abs/1106.1813, 2011. Disponível em: ⟨http://arxiv.org/abs/1106.1813⟩.

BREIMAN, L. Random forests. *Machine Learning*, Kluwer Academic Publishers, USA, v. 45, n. 1, p. 5–32, oct 2001. ISSN 1573-0565. Disponível em: ⟨https://doi.org/10.1023/A:1010933404324⟩.

CAREY, D.; ONG, K.-L.; WHITELEY, R.; CROSSLEY, K.; CROW, J.; MORRIS, M. Predictive modelling of training loads and injury in australian football. *International Journal of Computer Science in Sport*, v. 17, 06 2018.

CHANG, W.; CHENG, J.; ALLAIRE, J.; SIEVERT, C.; SCHLOERKE, B.; XIE, Y.; ALLEN, J.; MCPHERSON, J.; DIPERT, A.; BORGES, B. *shiny: Web Application Framework for R*. Boston, MA, 2025. R package version 1.11.1. Disponível em: ⟨https://CRAN.R-project.org/package=shiny⟩.

CHIU, S. L. Method and software for extracting fuzzy classification rules by subtractive clustering. In: *1996 Biennial Conference of the North American Fuzzy Information Processing Society (NAFIPS)*. Berkeley, CA, USA: IEEE, 1996. p. 461–465.

CONMEBOL. *265 milhões de pessoas jogam futebol no mundo inteiro*. 2013. Acesso em: 11 dez. 2025. Disponível em: ⟨https://www.conmebol.com/pt-br/notas-pt-br/265-milhoes-de-pessoas-jogam-futebol-no-mundo-inteiro/⟩.

CUSTÓDIO, I. J. O. *Análise quantitativa e qualitativa da carga de treinamento de uma equipe de futebol*. 74 p. Dissertação (Monografia (Especialização em Treinamento Esportivo)) — Universidade Federal de Minas Gerais (UFMG), Escola de Educação Física, Fisioterapia e Terapia Ocupacional, Belo Horizonte, 2011. Orientador: Leszek Szmuchrowski. Disponível em: ⟨https://repositorio.ufmg.br/items/1424749c-3606-4a20-a97b-846312bed583⟩.

EETVELDE, H. V.; MENDONÇA, L. D.; LEY, C.; SEIL, R.; TISCHER, T. Machine learning methods in sport injury prediction and prevention: a systematic review. *Journal of Experimental Orthopaedics*, Springer, v. 8, n. 1, p. 27, 2021. Disponível em: ⟨https://doi.org/10.1186/s40634-021-00346-x⟩.

EMERY, C. A.; PASANEN, K. Current trends in sport injury prevention. *Best Practice & Research Clinical Rheumatology*, Elsevier, v. 33, n. 1, p. 3–15, 2019. Disponível em: ⟨https://doi.org/10.1016/j.berh.2019.02.009⟩.

FAUDE, O.; KOCH, T.; MEYER, T. Straight sprinting is the most frequent action in goal situations in professional football. *Journal of Sports Sciences*, 2012.

FIFA. *Women's Football: Member Associations Survey Report 2023*. Zurich, 2023. Acesso em: 11 dez. 2025. Disponível em: ⟨https://inside.fifa.com/womens-football/member-associations-survey-report-2023⟩.

FLOR, F. A.; JAFELICE, R. S. M.; SILVA, J. W. Avaliação do risco de lesões em atletas de futebol feminino profissional via HyFIS. In: Universidade Federal de Uberlândia. *Anais do I Workshop de Matemática da UFU*. Uberlândia, MG, 2025.

FLOR, F. A.; JAFELICE, R. S. M.; SILVA, J. W.; SANTOS, C. A. R. *Indicação de Lesões em Atletas de Futebol Feminino Profissional via Aprendizado de Máquina*. 2025a. Pôster apresentado no 35º Colóquio Brasileiro de Matemática. IMPA - Rio de Janeiro.

FLOR, F. A.; JAFELICE, R. S. M.; SILVA, J. W.; SANTOS, T. R. T. Sistema neuro-fuzzy DENFIS na análise preditiva do risco de lesões em atletas de futebol feminino. In: Universidade Federal do Triângulo Mineiro. *Anais da XIV Semana da Matemática da UFTM*. Uberaba, MG, 2025b. ISSN: 2238-5908.

FOSTER, C. Monitoring training in athletes with reference to overtraining syndrome. *Medicine & Science in Sports & Exercise*, v. 30, n. 7, p. 1164–1168, jul 1998. Disponível em: ⟨https://doi.org/10.1097/00005768-199807000-00023⟩.

FOSTER, C.; FLORHAUG, J. A.; FRANKLIN, J.; GOTTSCHALL, L.; HROVATIN, L. A.; PARKER, S.; DOLESHAL, P.; DODGE, C. A new approach to monitoring exercise training. *Journal of Strength and Conditioning Research*, v. 15, n. 1, p. 109–115, feb 2001. Disponível em: ⟨https://doi.org/10.1519/00124278-200102000-00019⟩.

GABBETT, T. J. The development and application of an injury prediction model for noncontact, soft-tissue injuries in elite collision sport athletes. *J Strength Cond Res*, v. 24, n. 10, p. 2593–2603, 2010. Disponível em: ⟨https://doi.org/10.1519/JSC.0b013e3181f19da4⟩.

GABBETT, T. J. The training-injury prevention paradox: should athletes be training smarter and harder? *British Journal of Sports Medicine*, v. 50, n. 5, p. 273–280, mar 2016. Disponível em: ⟨https://doi.org/10.1136/bjsports-2015-095788⟩.

GABBETT, T. J. How much? How fast? How soon? Three simple concepts for progressing training loads to minimize injury risk and enhance performance. *Journal of Orthopaedic & Sports Physical Therapy*, v. 50, n. 10, p. 570–573, Oct 2020. Disponível em: ⟨https://doi.org/10.2519/jospt.2020.9256⟩.

GALLO, T. F.; CORMACK, S. J.; GABBETT, T. J.; LORENZEN, C. H. Pre-training perceived wellness impacts training output in australian football players. *Journal of Sports Sciences*, Taylor & Francis, v. 34, n. 15, p. 1445–1451, 2016. Disponível em: ⟨https://doi.org/10.1080/02640414.2015.1119295⟩.

GÉRON, A. *Hands-on Machine learning with Scikit-Learn and TensorFlow: concepts, tools, and techniques to build intelligent systems*. Sebastopol, CA: O'Reilly Media, 2017. ISBN 978-1491962299.

HARALDSDOTTIR, K.; SANFILIPPO, J.; MCKAY, L.; WATSON, A. M. Decreased sleep and subjective well-being as independent predictors of injury in female collegiate volleyball players. *Orthopaedic Journal of Sports Medicine*, v. 9, n. 9, 2021. Disponível em: ⟨https://doi.org/10.1177/23259671211029285⟩.

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. 2. ed. Springer, 2009. (Springer Series in Statistics). Second edition, 12th printing 2017. ISBN 978-0-387-84857-0. Disponível em: ⟨https://hastie.su.domains/ElemStatLearn/⟩.

HULIN, B. T.; GABBETT, T. J.; LAWSON, D. W.; CAPUTI, P.; SAMPSON, J. A. The acute:chronic workload ratio predicts injury: high chronic workload may decrease injury risk in elite rugby league players. *British Journal of Sports Medicine*, v. 50, n. 4, p. 231–236, feb 2016. Disponível em: ⟨https://doi.org/10.1136/bjsports-2015-094817⟩.

IMPELLIZZERI, F. M.; RAMPININI, E.; COUTTS, A. J.; SASSI, A.; MARCORA, S. M. Use of rpe-based training load in soccer. *Medicine & Science in Sports & Exercise*, v. 36, n. 6, p. 1042–1047, jun 2004.

IVARSSON, A.; JOHNSON, U.; ANDERSEN, M. B.; TRANAEUS, U.; STENLING, A.; LINDWALL, M. Psychosocial factors and sport injuries: meta-analyses for prediction and prevention. *Sports Medicine*, Springer, v. 47, n. 2, p. 353–365, 2017. Disponível em: ⟨https://doi.org/10.1007/s40279-016-0578-x⟩.

JAFELICE, R. S. M.; BARROS, L. C.; BASSANEZI, R. C. *Teoria dos Conjuntos Fuzzy com Aplicações*. 3. ed. São Carlos, SP: Sociedade Brasileira de Matemática Aplicada e Computacional (SBMAC), 2023. ISBN 978-85-8215-100-3.

JAMES, G.; WITTEN, D.; HASTIE, T.; TIBSHIRANI, R. *An Introduction to Statistical Learning: with Applications in R*. 2. ed. Springer, 2021. (Springer Texts in Statistics). ISBN 978-1-0716-1418-1. Disponível em: ⟨https://doi.org/10.1007/978-1-0716-1418-1⟩.

JANG, J. ANFIS: Adaptive-network-based fuzzy inference system. *IEEE Transactions on Systems, Man, and Cybernetics*, v. 23, n. 3, p. 665–685, May-June 1993. Disponível em: ⟨https://doi.org/10.1109/21.256541⟩.

JANG, J. R.; SUN, C.-T.; MIZUTANI, E. *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*. Upper Saddle River: Prentice Hall, 1997.

JASPERS, A.; KUYVENHOVEN, J. P.; STAES, F.; FRENCKEN, W. G. P.; HELSEN, W. F.; BRINK, M. S. Examination of the external and internal load indicators' association with overuse injuries in professional soccer players. *Journal of Science and Medicine in Sport*, v. 21, n. 6, p. 579–585, jun 2018.

JAUHIAINEN, S.; KAUPPI, J. P.; KROSSHAUG, T.; BAHR, R.; BARTSCH, J.; ÄYRÄMÖ, S. Predicting ACL injury using machine learning on data from an extensive screening test battery of 880 female elite athletes. *The American Journal of Sports Medicine*, v. 50, n. 11, p. 2917–2924, set. 2022.

JIANG, H.; KWONG, C.; OKUDAN KREMER, G.; PARK, W.-Y. Dynamic modelling of customer preferences for product design using denfis and opinion mining. *Advanced Engineering Informatics*, v. 42, p. 100969, 2019. ISSN 1474-0346. Disponível em: ⟨https://doi.org/10.1016/j.aei.2019.100969⟩.

JOHANSEN, H. D.; JOHANSEN, D.; KUPKA, T.; RIEGLER, M. A.; HALVORSEN, P. Scalable infrastructure for efcient real-time sports analytics. In: *Companion Publication of the 2020 International Conference on Multimodal Interaction, ICMI '20 Companion*. New York, NY, USA: Association for Computing Machinery, 2020. p. 230–234. Disponível em: ⟨https://doi.org/10.1145/3395035.3425300⟩.

KASABOV, N. Ecos: Evolving connectionist systems and the eco learning paradigm. 03 1999.

KASABOV, N.; SONG, Q. Denfis: Dynamic evolving neural-fuzzy inference system and its application for time-series prediction. *Fuzzy Systems, IEEE Transactions on*, v. 10, p. 144 – 154, 2002.

KUHN, M. Building predictive models in r using the caret package. *Journal of Statistical Software*, v. 28, n. 5, p. 1–26, 2008. Disponível em: ⟨https://www.jstatsoft.org/index.php/jss/article/view/v028i05⟩.

LAVALLÉE, L.; FLINT, F. The relationship of stress, competitive anxiety, mood state, and social support to athletic injury. *Journal of Athletic Training*, v. 31, n. 4, p. 296–299, Oct 1996.

LIAW, A.; WIENER, M. Classification and regression by randomforest. *R News*, v. 2, n. 3, p. 18–22, 2002. Disponível em: ⟨https://CRAN.R-project.org/doc/Rnews/⟩.

LUGHOFER, E. *Evolving Fuzzy Systems - Methodologies, Advanced Concepts and Applications*. 1st. ed. Berlin: Springer Publishing Company, 2011. ISBN 3642180868.

MCCULLOCH, W. S.; PITTS, W. A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, v. 5, n. 4, p. 115–133, 1943. Disponível em: ⟨https://doi.org/10.1007/BF02478259⟩.

MEEUWISSE, W. H.; TYREMAN, H.; HAGEL, B.; EMERY, C. A dynamic model of etiology in sport injury: the recursive nature of risk and causation. *Clinical Journal of Sport Medicine*, Lippincott Williams & Wilkins, v. 17, n. 3, p. 215–219, 2007.

MENDES, I.; COSTA JÚNIOR, P. P.; BERGO, L.; LEITE, D. Redes neuro-fuzzy evolutivas embarcadas em sistemas microcontrolados. In: *II Congresso Brasileiro de Sistemas Fuzzy*. Natal, RN: CBSF, 2012. p. 630–644.

MEYER, D.; DIMITRIADOU, E.; HORNIK, K.; WEINGESSEL, A.; LEISCH, F. *e1071: Misc Functions of the Department of Statistics, Probability Theory Group (Formerly: E1071), TU Wien*. Vienna, Austria, 2024. R package version 1.7-16. Disponível em: ⟨https://CRAN.R-project.org/package=e1071⟩.

MIDOGLU, C.; WINTHER, A. K.; BOEKER, M.; PETTERSEN, S. D.; PEDERSEN, S.; RAGAB, N.; KUPKA, T.; HICKS, S. A.; RANDERS, M. B.; JAIN, R.; DAGENBORG, H. J.; PETTERSEN, S. A.; JOHANSEN, D.; RIEGLER, M. A.; HALVORSEN, P. A large-scale multivariate soccer athlete health, performance, and position monitoring dataset. *Scientific Data*, v. 11, n. 553, p. 1–19, 2024.

MINSKY, M.; PAPERT, S. *Perceptrons: An Introduction to Computational Geometry*. Cambridge, MA: MIT Press, 1969. ISBN 978-0-262-63022-1.

Museu do Futebol. *Copa do Mundo de Futebol Feminino de 2011*. 2011. Centro de Referência do Futebol Brasileiro (CRFB). Acesso em: 11 dez. 2025. Disponível em: ⟨https://museudofutebol.org.br/crfb/eventos/617123/⟩.

OKADA, H. K. R.; NEVES, A. R. N.; SHITSUKA, R. Análise de Algoritmos de Indução de Árvores de Decisão. *Research, Society and Development*, Universidade Federal de Itajubá, Research, Society and Development, v. 8, n. 11, p. 01–15, 2019. Disponível em: ⟨https://doi.org/10.33448/rsd-v8i11.1473⟩.

PATEL, H.; PATEL, H. K.; SHARMA, A.; SRIVASTAVA, S. Design and development of a model for tennis elbow injury prediction and prevention using Artificial Neural Network (ANN) and Adaptive Neuro-Fuzzy Inference System (ANFIS) approaches. *BMC Musculoskeletal Disorders*, v. 26, n. 1, p. 916, out. 2025.

PIZARRO, J. O. Globalização e futebol feminino: o mercado de circulação das atletas nos mundiais FIFA e jogos olímpicos. *RBFF - Revista Brasileira de Futsal e Futebol*, v. 16, n. 66, p. 390–414, dez. 2024. Disponível em: ⟨https://www.rbff.com.br/index.php/rbff/article/view/1454⟩.

Posit team. *RStudio: Integrated Development Environment for R*. Boston, MA, 2025. Disponível em: ⟨http://www.posit.co/⟩.

PUTLUR, P.; FOSTER, C.; MISKOWSKI, J. A.; KANE, M. K.; BURTON, S. E.; SCHEETT, T. P.; MCGUIGAN, M. R. Alteration of immune function in women collegiate soccer players and college students. *Journal of Sports Science and Medicine*, v. 3, n. 4, p. 234–243, dec 2004.

R Core Team. *R: A Language and Environment for Statistical Computing*. Vienna, Austria, 2024. Disponível em: ⟨https://www.R-project.org/⟩.

REZENDE, S. *Sistemas inteligentes: fundamentos e aplicações*. Manole, 2003. ISBN 9788520416839. Disponível em: ⟨https://books.google.com.br/books?id=UsJe_PlbnWcC⟩.

RIZA, L. S.; BERGMEIR, C.; HERRERA, F.; BENÍTEZ, J. M. Learning from data using the r package "FRBS". In: *2014 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*. Beijing, China: IEEE, 2014. p. 2149–2155.

ROBERTS, S. S. H.; TEO, W.; WARMINGTON, S. A. Effects of training and competition on the sleep of elite athletes: a systematic review and meta-analysis. *British Journal of Sports Medicine*, BMJ, v. 53, p. 513–522, 2019.

ROGALSKI, B.; DAWSON, B.; HEASMAN, J.; GABBETT, T. J. Training and game loads and injury risk in elite australian footballers. *Journal of Science and Medicine in Sport*, v. 16, n. 6, p. 499–503, 2013. ISSN 1440-2440. Disponível em: ⟨https://www.sciencedirect.com/science/article/pii/S1440244012011334⟩.

ROSENBLATT, F. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, v. 65, n. 6, p. 386–408, 1958. Disponível em: ⟨https://doi.org/10.1037/h0042519⟩.

SAMUEL, A. L. Some studies in machine learning using the game of checkers. *IBM Journal of Research and Development*, IBM, v. 3, n. 3, p. 210–229, 1959. Disponível em: ⟨https://doi.org/10.1147/rd.33.0210⟩.

SAW, A. E.; MAIN, L. C.; GASTIN, P. B. Monitoring the athlete: the integration of subjective and objective measures. *Sports Medicine*, Springer, v. 46, n. 2, p. 171–186, 2016a.

SAW, A. E.; MAIN, L. C.; GASTIN, P. B. Monitoring the athlete training response: subjective self-reported measures trump commonly used objective measures: a systematic review. *British Journal of Sports Medicine*, BMJ Publishing Group Ltd, v. 50, n. 5, p. 281–291, 2016b.

SINGH, S.; GIRI, M. Comparative study id3, cart and c4.5 decision tree algorithm: A survey. *International Journal of Advanced Information Science and Technology (IJAIST)*, v. 3, n. 7, p. 47–52, 2014.

SOLIGARD, T.; SCHWELLNUS, M.; ALONSO, J.-M.; BAHR, R.; CLARSEN, B.; DIJKSTRA, H. P.; GABBETT, T.; GLEESON, M.; HÄGGLUND, M.; GARCÍA-MAS, A. *et al.* How much is too much? (part 1) international olympic committee consensus statement on load in sport and risk of injury. *British Journal of Sports Medicine*, BMJ Publishing Group Ltd, v. 50, n. 17, p. 1030–1041, 2016.

TAKAGI, T.; SUGENO, M. Fuzzy identification of systems and its applications to modeling and control. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-15, n. 1, p. 116–132, 1985.

TANGIRALA, S. Evaluating the impact of gini index and information gain on classification using decision tree classifier algorithm. *(IJACSA) International Journal of Advanced Computer Science and Applications*, Faculty of Business, University of Botswana, v. 11, n. 2, p. 47–52, 2020.

THE MATH WORKS INC. *Optimization Toolbox User's Guide*. Natick, MA, 1996. v. 3, 19-24 p.

THERNEAU, T. M.; ATKINSON, B. A. *rpart: Recursive Partitioning and Regression Trees*. [S.l.], 2024. R package version 4.1.23. Disponível em: ⟨https://CRAN.R-project.org/package=rpart⟩.

THORPE, R. T.; STRUDWICK, A. J.; BUCHHEIT, M.; ATKINSON, G.; DRUST, B.; GREGSON, W. Monitoring fatigue during the in-season competitive phase in elite soccer players. *International Journal of Sports Physiology and Performance*, Human Kinetics, v. 10, n. 8, p. 958–964, 2015.

WALSH, N. P.; HALSON, S. L.; SARGENT, C.; AL. et. Sleep and the athlete: narrative review and 2021 expert consensus recommendations. *British Journal of Sports Medicine*, BMJ, v. 55, p. 356–368, 2021.

WINDT, J.; GABBETT, T. J. How do training and competition workloads relate to injury? The workload-injury aetiology model. *British Journal of Sports Medicine*, v. 51, n. 5, p. 428–435, mar 2017.

WITTEN, I. H.; FRANK, E.; HALL, M. A. *Data Mining: Practical Machine Learning Tools and Techniques*. 3rd. ed. Burlington, MA: Morgan Kaufmann, 2011. ISBN 9780123748560.

YAGER, R.; FILEV, D. Generation of fuzzy rules by mountain clustering. *Journal of Intelligent and Fuzzy Systems*, v. 2, n. 3, p. 209–219, 1994. Disponível em: ⟨https://doi.org/10.3233/IFS-1994-2301⟩.

ZADEH, L. A. Fuzzy sets. *Information and Control*, v. 8, p. 338–353, 1965. Disponível em: ⟨https://doi.org/10.1016/S0019-9958(65)90241-X⟩.

ZHOU, Z. H. *Ensemble Methods: Foundations and Algorithms*. Boca Raton, FL: CRC Press, 2012. ISBN 9781439830031.

ZULQUES, D. *A resistência do futebol feminino no país do futebol*. 2023. Acesso em: 11 dez. 2025. Disponível em: ⟨https://iree.org.br/a-resistencia-do-futebol-feminino-no-pais-do-futebol/⟩.