

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO CEARÁ
PRÓ-REITORIA DE PESQUISA, PÓS-GRADUAÇÃO E INOVAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Reginaldo Pereira Fernandes Ribeiro

Classificação com Opção de Rejeição em Comitês

Fortaleza – CE

13 de setembro de 2024

Reginaldo Pereira Fernandes Ribeiro

Classificação com Opção de Rejeição em Comitês

Dissertação submetida ao Programa de Pós-Graduação em Ciência da Computação do Instituto Federal de Educação, Ciência e Tecnologia do Ceará, como requisito parcial para obtenção do grau de Mestre em Ciência da Computação.

Linha de Pesquisa: Inteligência Artificial

Orientador: Dr. Ajalmar Rego da Rocha Neto

Coorientador: Dr. Thiago Alves Rocha

Instituto Federal de Educação, Ciência e Tecnologia do Ceará (IFCE)

Pró-Reitoria de Pesquisa, Pós-Graduação e Inovação (PRPI)

Programa de Pós-Graduação em Ciência da Computação (PPGCC)

Fortaleza – CE

13 de setembro de 2024

Dados Internacionais de Catalogação na Publicação

Instituto Federal do Ceará - IFCE

Sistema de Bibliotecas - SIBI

Ficha catalográfica elaborada pelo SIBI/IFCE, com os dados fornecidos pelo(a) autor(a)

R484c Ribeiro, Reginaldo Pereira Fernandes.

CLASSIFICAÇÃO COM OPÇÃO DE REJEIÇÃO EM COMITÊS / Reginaldo Pereira
Fernandes Ribeiro. - 2018.

103 f. : il. color.

Dissertação (Mestrado) - Instituto Federal do Ceará, Mestrado em Ciência da
Computação, Campus Fortaleza, 2018.

Orientação: Prof. Dr. Ajalmar Rêgo da Rocha Neto.

Coorientação: Prof. Dr. Thiago Alves Rocha.

1. Confiabilidade. 2. Opção de Rejeição. 3. Índice de Gini. 4. Comitês. I. Título.

CDD 004



INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO CEARÁ
Av. 13 de maio, 2081 - Bairro Benfica - CEP 60040-531 - Fortaleza - CE - www.ifce.edu.br

FOLHA DE APROVAÇÃO

Processo: 23256.008345/2024-91

Interessado: Reginaldo Pereira Fernandes Ribeiro

REGINALDO PEREIRA FERNANDES RIBEIRO

CLASSIFICAÇÃO COM OPÇÃO DE REJEIÇÃO EM COMITÊS

Dissertação apresentada ao Curso de Pós-Graduação em Ciência da Computação do Instituto Federal de Educação, Ciência e Tecnologia do Ceará (IFCE) *Campus* de Fortaleza, como requisito parcial para obtenção do Título de Mestre em Ciência da Computação.

Aprovada em: **13/09/2024**.

BANCA EXAMINADORA

Prof. Dr. Ajalmar Rêgo da Rocha Neto (Orientador)
Instituto Federal de Educação, Ciência e Tecnologia do Ceará - PPGCC/IFCE

Prof. Dr. Ronaldo Fernandes Ramos
Instituto Federal de Educação, Ciência e Tecnologia do Ceará - IFCE

Prof. Dr. João Paulo Pordeus Gomes
Universidade Federal do Ceará – UFC



Documento assinado eletronicamente por **Ajalmar Rego da Rocha Neto, Professor do Ensino Básico, Técnico e Tecnológico**, em 16/09/2024, às 14:29, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por **Joao Paulo Gomes, Usuário Externo**, em 16/09/2024, às 15:03, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por **Ronaldo Fernandes Ramos, Professor do Ensino Básico, Técnico e Tecnológico**, em 16/09/2024, às 17:54, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade do documento pode ser conferida no site https://sei.ifce.edu.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0 informando o código verificador **6486306** e o código CRC **6F46F23A**.

Dedico este trabalho à memória de minha mãe, Clemencia Pereira Fernandes, e à minha avó, Evangelista Rodrigues Pereira da Silva, que com amor, força e sabedoria moldaram meu caminho. Agradeço profundamente por me ensinarem, desde cedo, o valor do conhecimento, da perseverança e da dignidade, valores que me conduziram até aqui e que levarei sempre comigo.

Agradecimentos

Gostaria de expressar minha sincera gratidão ao Instituto Federal do Ceará (IFCE) e ao Programa de Pós-Graduação em Ciência da Computação (PPGCC) pelo suporte fundamental ao longo da minha jornada de mestrado. A infraestrutura, os recursos e o ambiente acadêmico proporcionados foram essenciais para o desenvolvimento deste trabalho.

Aos meus colegas mestrandos Francisco Acélio Cunha de Sousa e Francisco Mateus Rocha Filho, agradeço pela colaboração constante e pelas discussões enriquecedoras que tanto contribuíram para a evolução deste projeto. Suas companhias tornaram essa trajetória mais leve e produtiva, transformando desafios em oportunidades de crescimento.

Um agradecimento especial aos meus orientadores, Dr. Ajalmar Rêgo da Rocha Neto e Dr. Thiago Alves Rocha, por suas orientações valiosas, conselhos esclarecedores e apoio incondicional. Suas expertises e dedicações foram cruciais para a realização deste estudo, guiando-me com sabedoria e encorajamento.

*“A maior glória em viver não está em nunca cair, mas em nos levantar a cada vez que
caímos.”*

(Nelson Mandela)

Resumo

Neste trabalho é proposto um método para aprimorar a confiabilidade nas classificações em comitês, baseando-se na utilização do Índice de Gini (IG) como um limite de confiança com a abordagem de opção de rejeição. Inicialmente, determinamos limiares de confiança para aceitar ou rejeitar as predições resultantes dos comitês de classificação, onde utilizamos o IG através da distribuição das probabilidades de cada classe. Em um experimento posterior, interferimos no prognóstico de modelos cuja função de decisão é baseada na média dos classificadores base. Dessa maneira, a confiabilidade do modelo será computada em função do grau de pureza das predições dos modelos base, utilizando a métrica IG como critério de pureza. Nos dois casos, aceitamos ou rejeitamos a classificação de acordo com o princípio estabelecido e calculamos o risco empírico a fim de esclarecer a relação entre a taxa de erro e a taxa de rejeição. Os resultados deste estudo mostram que o emprego de técnicas de opção de rejeição proporciona, de forma geral, maior confiabilidade aos comitês de classificadores. A definição de intervalos de valores do IG pode ser utilizada para determinar a rigidez ou permissividade do comitê. Assim, pode-se escolher, a depender do contexto, o intervalo de valores que forneça o critério de confiabilidade mais apropriado. A abordagem desse estudo revela um grande potencial para otimização de comitês de classificadores em uma variedade de aplicações, como detecção de fraudes, detecção de spam e diagnóstico médico.

Palavras-chave: Confiabilidade, Opção de Rejeição, Índice de Gini. Comitês.

Abstract

In this work, we propose a method to improve the reliability of ensemble classifications by using the Gini Index (GI) as a confidence threshold within the reject option approach. Initially, we determine confidence thresholds to accept or reject the predictions resulting from the classification ensembles, employing the GI through the probability distribution of each class. In a subsequent experiment, we intervene in the decision-making process of models whose decision function is based on the average of the base classifiers. In this way, the model's reliability is computed as a function of the purity degree of the base models' predictions, using the GI metric as the purity criterion. In both cases, we accept or reject the classification according to the established principle and calculate the empirical risk to clarify the relationship between the error rate and the rejection rate. The results of this study demonstrate that using reject option techniques generally provides greater reliability to classifier ensembles. The definition of GI value ranges can be used to determine the committee's strictness or permissiveness. Thus, depending on the context, one can choose the range of values that provides the most appropriate reliability criterion. The approach of this study shows great potential for optimizing classifier ensembles in various applications, such as fraud detection, spam detection, and medical diagnosis.

Keywords: Reliability. Rejection Option. Gini Index. Ensembles.

Lista de ilustrações

Figura 1 – Exemplo de classificação binária com opção de rejeição.	58
Figura 2 – Regiões de rejeição e aceitação da regra ótima de reconhecimento. . . .	60
Figura 3 – Curva de compromisso entre o erro e a rejeição.	61
Figura 4 – Curva hipotética de compromisso entre o erro e a rejeição.	61
Figura 5 – Classificação com Opção de Rejeição.	67
Figura 6 – Votação com Opção de Rejeição (VRO).	69
Figura 7 – Relação entre Taxas de Acurácia e Rejeição - BCW	77
Figura 8 – Relação entre Taxas de Acurácia e Rejeição - FISH	80
Figura 9 – Relação entre Taxas de Acurácia e Rejeição - IRIS	83
Figura 10 – Relação entre Taxas de Acurácia e Rejeição - MNIST	86
Figura 11 – Relação entre Taxas de Acurácia e Rejeição - VCP	89
Figura 12 – Relação entre Taxas de Acurácia e Rejeição - WINE	92

Lista de tabelas

Tabela 1 – Exemplo de Classificação com VRO	70
Tabela 2 – Resumo dos Conjuntos de Dados	73
Tabela 3 – Hiper-parâmetros para o XGB	74
Tabela 4 – Hiper-parâmetros para o MLP	74
Tabela 5 – Hiper-parâmetros para o SVC	75
Tabela 6 – Hiper-parâmetros para o K-NN	75
Tabela 7 – Hiper-parâmetros para o RF	75
Tabela 8 – Acurácia e Rejeição (ACC% e REJ%) em BCW	78
Tabela 9 – Rendimento dos classificadores para $w_r = 0.08$	78
Tabela 10 – Rendimento dos classificadores para $w_r = 0.29$	79
Tabela 11 – Rendimento dos classificadores para $w_r = 0.50$	79
Tabela 12 – Acurácia e Rejeição (ACC% e REJ%) em FISH	81
Tabela 13 – Rendimento dos classificadores para $w_r = 0.15$	81
Tabela 14 – Rendimento dos classificadores para $w_r = 0.44$	82
Tabela 15 – Rendimento dos classificadores para $w_r = 0.81$	82
Tabela 16 – Acurácia e Rejeição (ACC% e REJ%) em IRIS	84
Tabela 17 – Rendimento dos classificadores para $w_r = 0.11$	84
Tabela 18 – Rendimento dos classificadores para $w_r = 0.39$	85
Tabela 19 – Rendimento dos classificadores para $w_r = 0.67$	85
Tabela 20 – Acurácia e Rejeição (ACC% e TR%) em MNIST	87
Tabela 21 – Rendimento dos classificadores para $w_r = 0.15$	87
Tabela 22 – Rendimento dos classificadores para $w_r = 0.45$	88
Tabela 23 – Rendimento dos classificadores para $w_r = 0.82$	88
Tabela 24 – Acurácia e Rejeição (ACC% e REJ%) em VCP	90
Tabela 25 – Rendimento dos classificadores para $w_r = 0.11$	90
Tabela 26 – Rendimento dos classificadores para $w_r = 0.39$	91
Tabela 27 – Rendimento dos classificadores para $w_r = 0.67$	91
Tabela 28 – Acurácia e Rejeição (ACC $\pm$ REJ) em WINE	93
Tabela 29 – Rendimento dos classificadores para $w_r = 0.11$	93
Tabela 30 – Rendimento dos classificadores para $w_r = 0.39$	93
Tabela 31 – Rendimento dos classificadores para $w_r = 0.67$	94

Lista de Algoritmos

1	Classificador com Opção de Rejeição	64
2	RO_VOTING com Opção de Rejeição	65
3	Classificador com Opção de Rejeição (CRO) - Proposta 1	69
4	Votação com Opção de Rejeição (VRO)	70

Lista de abreviaturas e siglas

AI	Artificial Intelligence
IG	Índice de Gini
NN	Neural Network
MLP	Multilayer Perceptron
BRCW	Breast Cancer Wisconsin
IRIS	Iris
ML	Machine Learning
RO	Reject Option
SVC	Support Vector Classifier
VRTC	Vertebral Column
WINE	Wine

Lista de símbolos

$\mathcal{X}$	Pontos de dados
$\mathbf{Y}$	Rótulos
	item[$\mathbf{X}$] Conjunto de dados
$\mathcal{C}$	clusters
X_L	Dados rotulados
Y_L	Rótulos
X_U	Dados não rotulados
Y_U	Rótulos estimados
z_j^l	Somatório ponderado para o neurônio
w_{ij}^l	Peso da conexão entre o neurônio i na camada $l - 1$ e o neurônio j na camada l
a_i^{l-1}	Saída do neurônio i na camada $l - 1$
a_i^{l-1}	Saída do neurônio i na camada $l - 1$
b_j^l	é o viés (<i>bias</i>)

Sumário

1	INTRODUÇÃO	29
1.1	Motivação para a Dissertação	30
1.2	Descrição do problema	31
1.3	Objetivo Geral e Objetivos Específicos	32
1.4	Produção científica	33
1.5	Organização da Dissertação	33
2	FUNDAMENTAÇÃO TEÓRICA	35
2.1	Tipos de Aprendizado	35
2.1.1	Aprendizado Não Supervisionado	36
2.1.2	Aprendizado Semi-Supervisionado	36
2.1.3	Classificação de padrões	36
2.2	Classificadores	37
2.2.1	Perceptron Multicamadas (Multilayer Perceptron - MLP)	37
2.2.1.1	Algoritmo de Retropropagação do Erro	37
2.2.1.2	Funções de Ativação	37
2.2.1.3	Cálculo dos Pesos da Camada de Saída	39
2.2.1.4	Cálculo dos Gradientes Locais	39
2.2.1.5	Atualização dos Pesos	39
2.2.2	Máquinas de Vetores de Suporte (Support vector machine - SVM)	40
2.2.3	K-Vizinhos Mais Próximos (k-Nearest Neighbors, k-NN)	41
2.3	Naive Bayes Gaussiano (Gaussian Naive Bayes - GNB)	43
2.3.1	Definição Matemática	43
2.4	Aprendizado de Comitês	44
2.4.1	Combinação de componentes (Majority Voting)	44
2.4.1.1	Voto Majoritário Simples (Hard Voting)	44
2.4.1.2	Votação Suave (Soft Voting)	45
2.4.1.3	Votação Suave Ponderada (Soft Voting Average)	45
2.4.2	Boosting	46
2.4.2.1	Extreme Gradient Boosting (XGBoost)	46
2.4.2.2	Extreme Gradient Boosting (XGBoost)	47
2.4.3	Floresta Aleatória (Random Forest -RF)	48
2.4.3.1	Construção das Árvores de Decisão:	48
2.4.3.2	Votação Majoritária:	49
2.4.3.3	Divisões Baseadas no Índice de Gini:	49

2.5	Estimação dos Parâmetros	49
2.6	Estimativa de confiança	50
2.6.1	Regra de Decisão de Bayes	50
2.6.2	Teoria de Decisão Bayesiana e Critério Máximo a Posteriori (MAP)	51
2.6.2.1	Definição Formal	51
2.6.3	Probabilidades a Priori das Classes	52
2.6.4	Regra do vizinho mais próximo	52
2.6.5	Platt Scaling	53
2.6.6	Probabilidade Máxima de Classe (MCP)	53
2.6.7	Índice de Impureza de Gini	54
2.6.8	Definição do Índice de Gini	54
2.7	Conclusão	55
3	CLASSIFICAÇÃO COM OPÇÃO DE REJEIÇÃO	57
3.1	Definição Teórica	57
3.2	Regra de decisão ótima	58
3.3	Abordagens de Classificação Binária com Opção de Rejeição	62
3.3.1	Um único classificador binário padrão	62
3.3.2	Dois Classificadores Independentes	62
3.3.3	Classificador com Opção de Rejeição Incorporada	62
3.3.4	Classificação Multi-Classe com Opção de Rejeição	63
3.3.5	Classificadores com opção de rejeição	63
3.3.5.1	Regra de Decisão	63
3.3.6	Avaliação de Classificador com Opção de Rejeição	65
3.4	Conclusão	66
4	COMITÊS DE CLASSIFICADORES COM OPÇÃO DE REJEIÇÃO	67
4.1	Classificação com Opção de Rejeição (CRO) - Proposta 1	67
4.1.1	Geração de Saídas dos modelos	67
4.1.2	Tomada de Decisão com Opção de Rejeição	68
4.2	Comitê de classificadores com opção de rejeição	69
4.2.1	Votação com Opção de Rejeição (VRO) - Proposta 2	69
4.2.2	Geração das Saídas do Comitê	69
4.3	Conclusão	71
4.3.1	Classificação com Opção de Rejeição (CRO) - Proposta 1	71
4.3.2	Votação com Opção de Rejeição (VRO) - Proposta 2	71
5	RESULTADOS E DISCUSSÕES	73
5.1	Metodologia de Experimentos	73
5.1.1	Problemas de Classificação	73

5.1.2	Parâmetros de Classificação com Opção de Rejeição	75
5.1.3	Avaliação dos Modelos	76
5.2	Resultados	76
5.2.1	Conjunto de dados BCW	77
5.2.1.1	Desempenho na Curva AR em BCW	77
5.2.1.2	Análise dos Classificadores em Relação ao Risco Empírico em BCW	78
5.2.2	Conjunto de dados FISH	80
5.2.2.1	Desempenho na Curva AR em FISH	80
5.2.2.2	Análise dos Classificadores em Relação ao Risco Empírico em FISH	81
5.2.3	Conjunto de dados IRIS	83
5.2.3.1	Desempenho na Curva AR em IRIS	83
5.2.3.2	Análise dos Classificadores em Relação ao Risco Empírico em IRIS	84
5.2.4	Conjunto de dados MNIST	86
5.2.4.1	Desempenho na Curva AR em MNIST	86
5.2.4.2	Análise dos Classificadores em Relação ao Risco Empírico em MNIST	87
5.2.5	Conjunto de dados VCP	89
5.2.5.1	Desempenho na Curva AR em VCP	89
5.2.5.2	Análise dos Classificadores em Relação ao Risco Empírico em VCP	90
5.2.6	Conjunto de dados WINE	92
5.2.6.1	Desempenho na Curva AR em WINE	92
5.2.6.2	Análise dos Classificadores em Relação ao Risco Empírico em WINE	92
5.3	Análise geral dos Resultados	94
5.3.1	Análise comportamento curva AR	94
5.3.2	Análise com base no risco empírico (REMP)	95
6	CONCLUSÕES E TRABALHOS FUTUROS	97
6.1	Conclusões	97
6.2	Trabalhos Futuros	98
	REFERÊNCIAS	99

1 Introdução

A presença da inteligência artificial (IA) em atividades que necessitam de tomadas de decisões além da capacidade da inteligência humana é crescente. IA refere-se à capacidade de sistemas computacionais em realizar tarefas que tradicionalmente exigem inteligência humana, como reconhecimento de fala, tomada de decisões e tradução entre idiomas (RUSSELL; NORVIG, 2021). Esses sistemas são alimentados por grandes volumes de dados e algoritmos avançados que lhes permitem aprender, adaptar-se e melhorar ao longo do tempo, muitas vezes com desempenho superior ao dos humanos em tarefas específicas.

Dentro do campo da IA, o aprendizado de máquina (AM) é um dos principais pilares e refere-se ao estudo de algoritmos que aprendem a partir de dados para fazer previsões ou tomar decisões (BISHOP, 2016). O AM permite que sistemas ajustem seus comportamentos com base em experiências, aprimorando a precisão de suas ações com o tempo sem serem explicitamente programados para cada tarefa. O aprendizado de máquina pode ser categorizado em várias abordagens, sendo as mais comuns o aprendizado supervisionado e o não supervisionado. No aprendizado supervisionado, o modelo é treinado com dados rotulados, onde a relação entre as entradas e as saídas é aprendida para prever novas instâncias (GOODFELLOW; BENGIO; COURVILLE, 2016). Já no aprendizado não supervisionado, o modelo trabalha com dados não rotulados, buscando padrões ou estruturas intrínsecas nos dados (HASTIE; TIBSHIRANI; FRIEDMAN, 2009). Essas abordagens têm ampla aplicação em diversos campos, desde a análise de imagens até a previsão de séries temporais.

No aprendizado supervisionado, as duas principais tarefas são regressão e classificação. A regressão envolve prever um valor contínuo com base nas características de entrada, como a previsão de preços de imóveis (HASTIE; TIBSHIRANI; FRIEDMAN, 2009). Por outro lado, a classificação consiste em atribuir uma classe ou categoria a uma nova observação com base em dados de entrada, como a detecção de e-mails de spam (MURPHY, 2012). A precisão e a robustez são fundamentais em sistemas de classificação, especialmente quando aplicados a contextos sensíveis, como diagnósticos médicos ou sistemas de segurança.

No mundo real, sistemas inteligentes não devem apenas ser precisos, mas também devem ter a capacidade de reconhecer quando provavelmente estão incorretos e evitar tomadas de decisões erradas (CHOW, 1957; ZHANG et al., 2023). Em contextos como a detecção de fraudes financeiras, por exemplo, onde padrões sutis e evasivos visam confundir os modelos de classificação tradicionais (SOUSA; MORA; CARDOSO, 2009; ZHANG

et al., 2023), a capacidade de rejeição é fundamental. Ao empregar a rejeição, é possível detectar transações que estão nos limites da classificação correta, evitando erros críticos. Da mesma forma, em sistemas de diagnóstico médico baseados em aprendizado de máquina, a identificação de patologias pode ser desafiadora devido à complexidade das características clínicas (HASAN; JAWAD, 2022; LAROUÏ et al., 2021). A capacidade de rejeitar análises incertas ou ambíguas é crucial, pois pode impactar diretamente no tratamento dos pacientes. Além disso, a classificação com opção de rejeição tem se mostrado eficaz em setores como processamento de linguagem natural e análise de sentimentos (MENA; PUJOL; VITRIÀ, 2020), onde pode identificar e rejeitar mensagens com características equivocadas, evitando que material legítimo seja incorretamente filtrado.

A aplicação da classificação com opção de rejeição de padrões surge como uma abordagem alternativa para aprimorar a confiabilidade e a eficácia desses sistemas de aprendizado de máquina. Essa estratégia combina a especificação convencional com a capacidade de rejeitar amostras de difícil identificação, destacando-se em diversas áreas e demonstrando um potencial significativo para enfrentar desafios onde o ônus do erro é maior do que o custo de renunciar (ROCHA NETO et al., 2011). Portanto, a capacidade de rejeição não só melhora a precisão dos sistemas, mas também a confiança nas decisões tomadas, refletindo a importância de considerar a incerteza na avaliação de modelos de aprendizado de máquina.

Para contribuir com os estudos sobre rejeição de padrões, propomos uma solução com base na aplicação do Índice de Gini como uma estratégia para permitir uma melhor tomada de decisão em relação à confiabilidade das classificações realizadas por comitês. O método consiste em aplicar limites de confiança para rejeitar a tomada de decisão em comitês de classificadores baseados na distribuição da pureza dos resultados obtidos no processo de previsão do comitê. O IG é um indicador que avalia a incerteza das predições. Originalmente utilizado para medir a desigualdade em distribuições, o IG é adaptado neste contexto para avaliar a confiança de um modelo em suas classificações (KUAN, 2003; YUAN; WU; ZHANG, 2021).

1.1 Motivação para a Dissertação

A confiabilidade é uma métrica importante para modelos que usam abordagens baseadas em rejeição. Essas abordagens têm vantagens significativas em várias áreas, como detecção de fraudes, direção autônoma, diagnóstico médico, sistemas de recomendação e análise de sentimento (ZHANG et al., 2023). A confiabilidade das classificações desempenha um papel crucial no aumento da segurança e da precisão em vários cenários.

Para minimizar os possíveis erros obtidos por classificação de instâncias, os comitês surgem como uma abordagem alternativa para refinar a precisão das classificações. Esses

modelos combinam a predição de vários modelos individuais, aproveitando a diversidade para obter um resultado mais robusto e geralmente mais específico. Entre os comitês de classificadores amplamente utilizados, destacam-se os das abordagens *Voting*, *Boosting* e *Random Forest* (MIENYE; SUN, 2022a).

Mesmo com a popularização dos métodos de comitê de classificadores, a possibilidade de erros de classificação é comum, já que no mundo real, as variáveis relacionadas aos problemas de aprendizado de máquina podem influenciar negativamente na tomada de decisão. Nesse contexto, a abordagem de classificação com opção de rejeição (CHOW, 1970), resultou em diversos estudos investigativos sobre o uso dessa técnica para estimar a confiança em problemas de classificação com rejeição.

1.2 Descrição do problema

Os sistemas de tomada de decisão nas diversas áreas de atuação de IA podem encontrar situações com dados novos desafiando os problemas tradicionais. Independente do modelo, erros de predição são inevitáveis, não importa o quão preciso seja. Os padrões e dados novos necessitam de estratégias que possibilitem que o classificador se omita de especificar erroneamente a amostra, obedecendo a custos limitantes que podem variar de acordo com o problema.

Vários estudos têm explorado a aplicabilidade da regra de decisão de Bayes em problemas de classificação com rejeição. Fukunaga (FUKUNAGA, 1990) propôs essa abordagem, cuja confiança é estimada com base na probabilidade a posteriori de pertencer a cada classe.

Outra abordagem relevante utiliza a Probabilidade Máxima a Posteriori (MAP), tal como a proposta de Chow (1970). Nesse contexto, diversos trabalhos têm investigado o uso dessa técnica para estimar a confiança em problemas de classificação com rejeição através do compromisso entre taxa de erro e rejeição.

Em relação à rejeição de valores atípicos, técnicas baseadas em análise destes padrões têm sido apuradas (HENDRICKX et al., 2021) e apresentadas ao longo dos anos. Em um estudo recente, Müller et al. (2023) analisam classificadores estabelecidos e robustos ao seu domínio, identificando regiões de competência na generalização do modelo.

No contexto de métricas de espaço, pesquisas exploraram o uso das distâncias Euclidiana e de Mahalanobis para avaliar a confiança em problemas de classificação com rejeição. Por exemplo, em k-vizinhos mais próximos, um limite de longitude entre eles pode ser usado para rejeitar amostras muito distantes (DUBUISSON; MASSON, 1993; HELLMAN, 1970).

Por fim, em relação às redes neurais, de acordo com Cordella et al. (1997) e Villon

et al. (2020), a confiança pode ser definida como a probabilidade do neurônio vencedor ou a diferença entre os dois neurônios principais. Essas abordagens possibilitaram a rejeição de amostras com baixa confiança, alcançando resultados promissores em termos de precisão e robustez do sistema.

Outros estudos exploraram estratégias de rejeição em classificadores em comitê (GANAIE et al., 2021). A aprendizagem em conjunto é uma abordagem que permite estimar a incerteza combinando diferentes modelos.

Neste contexto, para esta dissertação, investigamos a hipótese de que a inclusão da classificação com opção de rejeição com base no índice de impureza de Gini, a abordagem proporcionará confiabilidade no processo de tomada de decisão em comitês de classificadores. Através de simulações utilizando o modelo *Voting* por maioria simples, avaliamos a acurácia, a taxa de rejeição, a taxa de erro e o risco empírico das classificações obtidas. Além disso, analisamos graficamente a relação entre a taxa de erro e a taxa de rejeição, bem como comparação dos resultados com abordagens relevantes. Esperamos que esse estudo traga compreensão de causa e efeito sobre a eficácia da abordagem e seu impacto no processo de tomada de decisão realizada pelo comitê de classificadores com opção de rejeição.

1.3 Objetivo Geral e Objetivos Específicos

O objetivo geral deste trabalho é realizar um estudo comparativo entre métodos de classificação com opção de rejeição em problemas multiclasse, apresentando uma abordagem com base no índice de Gini (GI) como limiar de rejeição, tanto em comitês de classificadores quanto em modelos individuais. O estudo visa aprimorar a confiabilidade e a eficácia das decisões de classificação em cenários que apresentam padrões de difícil distinção.

Os objetivos específicos deste trabalho são:

- Investigar e desenvolver técnicas de rejeição aplicadas a problemas multiclasse, avaliando seu desempenho em diferentes cenários, identificando as vantagens, desafios e resultados alcançados por meio dessas estratégias, comparando abordagens em modelos individuais e em comitês de classificadores.
- Propor e desenvolver metodologias de rejeição que ajustem os limiares de rejeição com base nas características dos dados e na natureza do problema, buscando otimizar a confiabilidade das decisões de classificação, tanto em contextos de modelos individuais quanto em comitês de classificadores.
- Avaliar empiricamente a eficácia das estratégias de rejeição propostas por meio de experimentos e análises quantitativas, considerando métricas de desempenho como precisão, taxa de rejeição e taxa de erro, e comparando essas métricas com as obtidas em abordagens tradicionais, em ambos os cenários de classificação

(modelos individuais e comitês de classificadores).

1.4 Produção científica

No decorrer deste projeto de mestrado, os seguintes trabalhos científicos foram aprovados e disponibilizados para publicação:

- **Ribeiro, R. P. F., Rocha, T. A., Rocha Neto, A. R. Diagnóstico de patologias da coluna vertebral utilizando ensemble com opção de rejeição.** Trabalho aceito para apresentação no *XX Congresso Brasileiro de Informática em Saúde - CBIS'24*, 2024.
- Rocha Filho, F. M., Rocha, T. A., **Ribeiro, R. P. F.**, Rocha Neto, A. R. **Logic-based explanations for linear support vector classifiers with reject option.** In: NALDI, M. C.; BIANCHI, R. A. C. (Ed.). *Intelligent Systems*. Cham: Springer Nature Switzerland, 2023. p. 144–159. ISBN 978-3-031-45368-7.
- **Ribeiro, R. P. F., Rocha, T. A., Rocha Neto, A. R. Diagnóstico de patologias da coluna vertebral utilizando ensemble com opção de rejeição.** Trabalho aceito para publicação na revista *Journal of Health Informatics* (J. Health Inform.), versão estendida.

1.5 Organização da Dissertação

Esta dissertação está organizada da seguinte maneira:

- O Capítulo 2 aborda a introdução sobre classificadores, a descrição do processo de construção de comitês de classificadores, um resumo das estimativas de confiança e o conceito do Índice de Gini;
- O Capítulo 3 descreve a abordagem de Classificação com opção de Rejeição;
- O Capítulo 4 introduz a proposta de classificação com opção de rejeição em comitês;
- O Capítulo 5 descreve as metodologias usadas para construção e avaliação dos comitês de classificadores. Também são apresentados os resultados;
- O Capítulo 6 apresenta a conclusão e os trabalhos futuros.

2 Fundamentação Teórica

No contexto da aprendizagem de máquina e da análise estatística, destacam-se três conceitos interligados de significativa relevância: classificação de dados, construção de comitês e avaliação de estimativas de confiança. A tarefa de classificação, que atribui rótulos a dados, é fundamental para diversas aplicações, abrangendo desde diagnósticos médicos até detecção de fraudes. A partir dessa base, surgem os comitês, os quais combinam múltiplos modelos visando aprimorar o desempenho e a robustez das previsões. Paralelamente, a confiabilidade das predições desempenha um papel crucial, impulsionando a exploração de métodos que não apenas classificam, mas também quantificam a confiança associada.

Este capítulo está estruturado em aspectos fundamentais da classificação, comitês e estimativas de confiança. A seção 2.1 apresenta uma análise das técnicas de classificação, abrangendo os algoritmos tradicionais. Na seção 2.4, exploramos a importância dos comitês, destacando como a combinação estratégica de modelos individuais pode resultar em melhorias significativas no desempenho de predição. Já na seção 2.6, investigamos em detalhes os métodos de estimativa de confiança, desde a regra de decisão de Bayes até abordagens baseadas em probabilidades a priori e técnicas de vizinho mais próximo. Por fim, na subseção 2.6.7, concentramos nossa atenção no índice de impureza de Gini como métrica de confiabilidade.

2.1 Tipos de Aprendizado

O Reconhecimento de Padrões (RP) e a Aprendizagem de Máquina (AM) são disciplinas que desempenham papéis fundamentais na análise e interpretação de dados complexos. Com raízes na estatística, matemática e ciência da computação, tanto o RP quanto a AM buscam identificar regularidades, estruturas e tendências inerentes a conjuntos de dados, abrangendo uma ampla gama de aplicações, como visão computacional, processamento de sinais, biometria, medicina e muitos outros. Ambos desempenham papéis cruciais na tomada de decisões baseadas em dados (BISHOP, 2016).

As abordagens empregadas nestas disciplinas variam desde técnicas estatísticas tradicionais até métodos avançados baseados em aprendizado de máquina, como redes neurais, máquinas de vetores de suporte e árvores de decisão. Neste contexto, busca-se tanto a identificação de padrões e estruturas em dados quanto o desenvolvimento de algoritmos que podem aprimorar o desempenho em tarefas específicas através da análise de dados. (THEODORIDIS et al., 2010).

No contexto dessas disciplinas podemos elencar três principais abordagens de clas-

sificação: aprendizado não supervisionado, aprendizado semi-supervisionado e aprendizado supervisionado.

2.1.1 Aprendizado Não Supervisionado

No aprendizado não supervisionado, os dados não possuem rótulos ou categorias predefinidos, ou seja, são dados não rotulados. O objetivo é identificar padrões e agrupar os dados com base nas suas características internas, sem a necessidade de resultados esperados durante o treinamento. O algoritmo explora os dados para descobrir possíveis relacionamentos e regularidades, segmentando-os em grupos (*clusters*) com base em similaridades (TAN et al., 2019, p. 535).

Um exemplo típico de algoritmo utilizado para essa tarefa é o *k*-Médias. O *k*-Médias busca identificar os centróides dos clusters de forma a minimizar a soma das distâncias euclidianas entre os pontos e seus centróides (MANTHEY; RÖGLIN, 2009). A formulação matemática desse processo é expressa por um conjunto de n pontos de dados $\mathcal{X} \subseteq \mathbb{R}^d$, o objetivo do *k*-Médias é encontrar uma clusterização $\mathcal{C}_1, \dots, \mathcal{C}_k$ de $\mathcal{X}$, ou seja, uma partição de $\mathcal{X}$, bem como centros de *clusters* $\mathcal{C} = \{\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_k\}$, onde k é o número de *clusters* desejado.

2.1.2 Aprendizado Semi-Supervisionado

Na classificação semi-supervisionada, parte dos dados possui rótulos conhecidos, enquanto outra parte não possui rótulos. O objetivo é aproveitar as informações disponíveis para inferir rótulos para os dados não rotulados. Um método comum é a “*propagação de rótulo*”, onde a informação dos dados rotulados é usada para estimar rótulos para os dados não rotulados (ENGELEN; HOOS, 2020, p. 381). Isso é modelado através de funções de similaridade e penalidades para diferenças nos rótulos mostrada a seguir:

Dado um conjunto de dados rotulados $\mathbf{X}_L = \{\mathbf{x}_{L1}, \mathbf{x}_{L2}, \dots, \mathbf{x}_{Lm}\}$ com rótulos correspondentes $\mathbf{Y}_L = \{\mathbf{y}_{L1}, \mathbf{y}_{L2}, \dots, \mathbf{y}_{Lm}\}$ e um conjunto de dados não rotulados $\mathbf{X}_U = \{\mathbf{x}_{U1}, \mathbf{x}_{U2}, \dots, \mathbf{x}_{Un}\}$, o objetivo é estimar $\mathbf{Y}_U = \{\mathbf{y}_{U1}, \mathbf{y}_{U2}, \dots, \mathbf{y}_{Un}\}$ para os dados não rotulados.

2.1.3 Classificação de padrões

No aprendizado supervisionado, cada entrada de dados é associada a um rótulo conhecido e o objetivo é obter (treinar) um modelo para mapear as características dessas entradas para suas respectivas categorias (LAROUÏ et al., 2021). Esse processo utiliza um conjunto de dados rotulado, composto por pares ordenados de entrada e saída desejada. Durante o treinamento, o algoritmo aprende a partir desses exemplos rotulados para

desenvolver uma regra geral que mapeie as entradas para as saídas esperadas, resultando em um modelo (classificador) que faz previsões com base nos dados de entrada.

Dado um conjunto de treinamento $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$ com rótulos correspondentes $\mathbf{Y} = \{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_n\}$, o objetivo é aprender uma função $f : \mathbf{X} \rightarrow \mathbf{Y}$ que minimize o erro de classificação.

2.2 Classificadores

Nesta seção, vamos descrever alguns modelos de classificação envolvendo abordagens que serão utilizadas neste trabalho.

2.2.1 Perceptron Multicamadas (Multilayer Perceptron - MLP)

O Perceptron Multilayer (MLP) é uma rede neural artificial composta por camadas de entrada, ocultas e de saída, permitindo a modelagem de relações não lineares complexas. Ao contrário do perceptron de camada única, o MLP utiliza múltiplas camadas e funções de ativação não lineares para aprender representações abstratas e realizar classificações complexas. Cada neurônio em uma camada é densamente conectado aos neurônios da próxima camada, e o treinamento é feito por retropropagação do erro, ajustando os pesos para minimizar a função de perda.

2.2.1.1 Algoritmo de Retropropagação do Erro

O algoritmo de retropropagação do erro é uma técnica de otimização usada para treinar redes neurais, ajustando os pesos da rede com base no erro de predição. O processo é dividido em duas fases principais: a propagação para frente e a retropropagação do erro.

1. **Propagação para Frente:** Nesta fase, uma entrada é fornecida à rede e propagada através das camadas, passando pelos neurônios e suas respectivas funções de ativação, até a camada de saída. A saída da rede é comparada com a saída desejada e o erro é calculado.
2. **Retropropagação do Erro:** Na retropropagação, o erro calculado é propagado de volta através da rede para ajustar os pesos. Esta fase é crucial para calcular os gradientes necessários para a atualização dos pesos.

2.2.1.2 Funções de Ativação

As funções de ativação introduzem não-linearidade na rede neural, permitindo que o MLP aprenda e represente padrões complexos. Nas camadas ocultas, as ativações dos

neurônios são calculadas aplicando uma função de ativação ao somatório ponderado das entradas (Equação 2.1).

$$z_j^l = \sum_i w_{ij}^l \cdot a_i^{l-1} + b_j^l \quad (2.1)$$

onde:

- z_j^l é o somatório ponderado para o neurônio j na camada l .
- w_{ij}^l é o peso da conexão entre o neurônio i na camada $l - 1$ e o neurônio j na camada l .
- a_i^{l-1} é a saída do neurônio i na camada $l - 1$.
- b_j^l é o viés (*bias*) associado ao neurônio j na camada l .

As funções de ativação mais comumente utilizadas incluem a Sigmoid (Equação 2.2), que mapeia as saídas para o intervalo entre 0 e 1, e a ReLU (Equação 2.3), que aplica uma transformação linear retificada.

$$\sigma(z_j^l) = \frac{1}{1 + e^{-z_j^l}} \quad (2.2)$$

$$f(z_j^l) = \max(0, z_j^l) \quad (2.3)$$

Cada neurônio na camada l calcula a saída a_j^l aplicando a função Sigmoid ou ReLU ao somatório ponderado das entradas z_j^l .

Além dessas, outras funções de ativação também podem ser utilizadas, como a Tangente Hiperbólica:

$$\tanh(z_j^l) = \frac{e^{z_j^l} - e^{-z_j^l}}{e^{z_j^l} + e^{-z_j^l}} \quad (2.4)$$

ou a Leaky ReLU:

$$\text{Leaky ReLU}(z_j^l) = \begin{cases} z_j^l, & \text{se } z_j^l > 0 \\ \alpha z_j^l, & \text{se } z_j^l \leq 0 \end{cases} \quad (2.5)$$

Essas funções ajudam a adaptar as respostas da rede conforme a tarefa específica, determinando se um neurônio deve ser ativado ou não, dependendo dos sinais que recebe. A escolha da função de ativação pode impactar significativamente o desempenho e a capacidade de aprendizado do MLP.

2.2.1.3 Cálculo dos Pesos da Camada de Saída

Para calcular os pesos da camada de saída, utilizamos a seguinte equação:

$$u_k(t) = \sum_{i=0}^q m_{ik}(t) u_i(t) \quad (2.6)$$

onde $m_{ik}(t)$ são os pesos da camada de saída conectando os neurônios da camada anterior aos neurônios da camada de saída. Aqui, k varia de 1 a m , onde m é o número de neurônios na camada de saída. Após calcular o somatório ponderado, a saída é obtida aplicando uma função de ativação apropriada, como por exemplo a função softmax, particularmente utilizada em problemas de classificação multiclasse, pois transforma o vetor de saídas em uma distribuição de probabilidade. A função softmax é definida por:

$$a_k(t) = \frac{e^{u_k(t)}}{\sum_j e^{u_j(t)}} \quad (2.7)$$

2.2.1.4 Cálculo dos Gradientes Locais

Os gradientes locais dos neurônios da camada de saída são calculados usando a fórmula:

$$\delta_k(t) = e_k(t) \phi'[u_k(t)] \quad (2.8)$$

onde $e_k(t)$ é o erro entre a saída desejada e a saída gerada pelo modelo, e $\phi'[u_k(t)]$ é a derivada da função de ativação aplicada. Esse gradiente é então usado para atualizar os pesos da camada de saída.

Para calcular os gradientes locais dos neurônios da camada oculta, usamos:

$$\delta_i(t) = \phi'[u_i(t)] \sum_{k=1}^m m_{ki}(t) \delta_k(t) \quad (2.9)$$

onde $\delta_i(t)$ é o gradiente do neurônio i na camada oculta, e $m_{ki}(t)$ são os pesos da camada oculta.

2.2.1.5 Atualização dos Pesos

Após calcular os gradientes, a atualização dos pesos é realizada para minimizar a função de perda. A fórmula para atualizar os pesos da camada oculta é:

$$w_{ij}(t+1) = w_{ij}(t) + \eta \delta_i(t) x_j(t) \quad (2.10)$$

e para os pesos da camada de saída:

$$m_{ki}(t+1) = m_{ki}(t) + \eta \delta_k(t) y_j(t) \quad (2.11)$$

onde η é a taxa de aprendizado, w_{ij} são os pesos da camada oculta, $x_j(t)$ são as entradas para a camada oculta, m_{ki} são os pesos da camada de saída e $y_j(t)$ são as saídas da camada oculta.

2.2.2 Máquinas de Vetores de Suporte (Support vector machine - SVM)

Na formulação do problema de otimização a ser resolvido para treinar máquinas de vetores de suporte, o objetivo é maximizar a margem de separação $\rho = \frac{c}{\|\mathbf{w}\|^2}$ entre as classes, onde c é uma constante; ou, de forma equivalente, minimizar $\|\mathbf{w}\|^2$, onde $\mathbf{w}$ é o vetor de pesos da função de decisão:

$$f(\mathbf{x}) = \text{sign}(\mathbf{w}^T \mathbf{x} + b), \quad (2.12)$$

onde

$$\text{sign}(u) = \begin{cases} +1, & \text{se } u \geq 0 \\ -1, & \text{se } u < 0 \end{cases}$$

e b é o viés (CORTES; VAPNIK, 1995).

O problema primal para máquinas de vetores de suporte com margem suave é dado por:

$$\begin{aligned} \text{minimizar} \quad & \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^n \xi_i \\ \text{sujeito a} \quad & y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1 - \xi_i, \quad \forall i = 1, \dots, n, \\ & \xi_i \geq 0, \quad \forall i = 1, \dots, n. \end{aligned}$$

onde C é uma constante de regularização entre os erros permitidos e a maximização da margem de separação entre as classes, e ξ_i é a i -ésima variável de folga utilizada para permitir a ocorrência de erros de classificação.

O problema dual para máquinas de vetores de suporte com margem suave é definido como:

$$\text{maximizar} \left\{ \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) \right\} \quad (2.13)$$

$$\text{sujeito a} \quad \sum_{i=1}^N \alpha_i y_i = 0 \quad (2.14)$$

$$0 \leq \alpha_i \leq C \quad (2.15)$$

onde $K(\mathbf{x}_i, \mathbf{x}_j) = K_{ij} = \mathbf{x}_i^T \mathbf{x}_j$ para uma SVM linear, α_i é o i -ésimo multiplicador de Lagrange (BOSER; GUYON; VAPNIK, 1992). Cada multiplicador de Lagrange α_i nas máquinas de vetores de suporte está associado ao seu i -ésimo padrão de treinamento $\mathbf{x}_i$.

As condições de otimalidade para o problema dual são as seguintes:

$$\mathbf{w} = \sum_{i=1}^N \alpha_i \mathbf{x}_i \quad (2.16)$$

e

$$\sum_{i=1}^N \alpha_i y_i = 0. \quad (2.17)$$

Com base na Eq.(2.16), a função de decisão apresentada na Eq.(2.12) para SVMs pode ser mostrada da seguinte forma:

$$f(\mathbf{x}) = \text{sign}(g(\mathbf{x})) \quad (2.18)$$

onde

$$g(\mathbf{x}) = \sum_{i=1}^N \alpha_i K(\mathbf{x}_i, \mathbf{x}) + b. \quad (2.19)$$

onde $\mathbf{x}$ é o padrão a ser classificado, $\mathbf{x}_i$ é o i -ésimo padrão de treinamento; enquanto $K(\mathbf{x}_i, \mathbf{x}) = \mathbf{x}_i^T \mathbf{x}$ para uma SVM linear. Outro ponto interessante a destacar é que:

$$d(\mathbf{x}_i) = \frac{|g(\mathbf{x}_i)|}{\|\mathbf{w}\|} \quad (2.20)$$

representa a distância de qualquer padrão $\mathbf{x}_i$ ao hiperplano ótimo $\mathbf{w}$. Assim, $|g(\mathbf{x}_i)|$ também pode ser considerado um tipo de medida de distância, uma vez que o valor de $\|\mathbf{w}\|$ é o mesmo para todos os padrões de treinamento (ROCHA NETO et al., 2011).

2.2.3 K-Vizinhos Mais Próximos (k-Nearest Neighbors, k-NN)

O algoritmo dos k-vizinhos mais próximos (k-NN) é um método de aprendizado supervisionado não paramétrico amplamente utilizado na classificação de dados. Ele foi inicialmente desenvolvido por Evelyn Fix e Joseph Hodges, e posteriormente expandido por Thomas Cover (FIX; HODGES, 1989; COVER; HART, 1967). O k-NN é um exemplo de *lazy learner*, pois não constrói explicitamente um modelo durante a fase de treinamento; ao invés disso, adia o processo de modelagem até que uma instância de teste precise ser classificada (TAN et al., 2019, p. 223).

Para realizar a predição, o k-NN classifica uma nova instância $\mathbf{x}'$ comparando-a com todas as instâncias $\mathbf{x}_i$ de um conjunto de treinamento previamente armazenado. A similaridade entre $\mathbf{x}'$ e cada $\mathbf{x}_i$ é medida usando uma métrica de distância, como a distância

euclidiana ou a distância de Manhattan. A escolha da métrica depende do contexto e da natureza dos dados. A distância euclidiana entre duas instâncias $\mathbf{x}'$ e $\mathbf{x}_i$ é expressa como:

$$d(\mathbf{x}', \mathbf{x}_i) = \sqrt{\sum_{j=1}^d (\mathbf{x}'_j - \mathbf{x}_{ij})^2}, \quad (2.21)$$

em que d representa o número de atributos de cada instância.

Após calcular as distâncias, o algoritmo identifica os k vizinhos mais próximos, ou seja, os k exemplos de treinamento com as menores distâncias em relação à instância de teste $\mathbf{x}'$. O rótulo da classe prevista para $\mathbf{x}'$, denotada por y' , é determinada através de uma votação majoritária entre os k vizinhos. A classe mais frequente entre os vizinhos é escolhida como a classe predita, formalmente expressa por:

$$y' = \arg \max_u \sum_{(\mathbf{x}_i, y_i) \in D_k} \mathbb{I}(u = y_i), \quad (2.22)$$

em que u é uma classe candidata, D_k é o conjunto dos k vizinhos mais próximos, y_i é a classe do i -ésimo vizinho, e $\mathbb{I}(\cdot)$ é a função indicadora que retorna 1 se a condição for verdadeira e 0 caso contrário (TAN et al., 2019, p. 225).

Para melhorar a precisão do k-NN, especialmente em casos com ruído ou *outliers*¹, é comum utilizar uma ponderação baseada na distância, onde vizinhos mais próximos têm maior peso na votação. Nesse caso, a classe predita é determinada por:

$$y' = \arg \max_u \sum_{(\mathbf{x}_i, y_i) \in D_k} w_i \cdot \mathbb{I}(u = y_i), \quad (2.23)$$

onde o peso w_i de cada vizinho $\mathbf{x}_i$ é calculado como:

$$w_i = \frac{1}{d(\mathbf{x}', \mathbf{x}_i)^2}. \quad (2.24)$$

Essa ponderação diminui a influência de exemplos de treinamento mais distantes, que podem ser menos relevantes para a classificação da nova instância.

A escolha do parâmetro k é crucial para o desempenho do modelo k-NN. Valores pequenos de k podem levar ao *overfitting*, tornando o modelo sensível a *outliers* e ruídos. Por outro lado, valores grandes de k podem resultar em *underfitting*, ao incluir exemplos distantes que não representam adequadamente a vizinhança local da instância de teste. Assim, a seleção apropriada de k é essencial e costuma ser realizada através de técnicas como busca em grade com validação cruzada (TAN et al., 2019, p. 226).

¹ Outliers são amostras que estão muito longe do resto das amostras em sua classe.

2.3 Naive Bayes Gaussiano (Gaussian Naive Bayes - GNB)

O classificador Naive Bayes é um algoritmo de aprendizado supervisionado baseado no teorema de Bayes, que assume que as variáveis de entrada são independentes entre si, dado a classe alvo (KUNCHEVA, 2006). No caso específico do Naive Bayes Gaussiano, o algoritmo assume que as características contínuas de cada classe seguem uma distribuição normal (Gaussiana) (ZHANG, 2004). Essa versão do Naive Bayes é amplamente utilizada em problemas onde as variáveis de entrada são contínuas e se espera que elas sigam uma distribuição normal (BISHOP, 2016, p. 24).

2.3.1 Definição Matemática

O teorema de Bayes define a probabilidade posterior $P(C_k|\mathbf{x})$ de uma classe C_k dada uma amostra $\mathbf{x}$ como:

$$P(C_k|\mathbf{x}) = \frac{P(\mathbf{x}|C_k) \cdot P(C_k)}{P(\mathbf{x})}, \quad (2.25)$$

em que:

- $P(C_k|\mathbf{x})$ é a probabilidade posterior da classe C_k dado o vetor de características $\mathbf{x}$.
- $P(\mathbf{x}|C_k)$ é a probabilidade de $\mathbf{x}$ dado que a classe é C_k .
- $P(C_k)$ é a probabilidade a priori da classe C_k .
- $P(\mathbf{x})$ é a probabilidade marginal de $\mathbf{x}$.

Para o classificador Naive Bayes Gaussiano, assume-se que $P(\mathbf{x}|C_k)$ segue uma distribuição Gaussiana multivariada, que pode ser expressa como o produto das distribuições Gaussianas de cada característica assumindo independência entre elas:

$$P(\mathbf{x}|C_k) = \prod_{j=1}^n P(x_j|C_k), \quad (2.26)$$

onde n é o número de características em $\mathbf{x}$. Cada variável x_j é modelada como uma distribuição Gaussiana:

$$P(x_j|C_k) = \frac{1}{\sqrt{2\pi\sigma_{jk}^2}} \exp\left(-\frac{(x_j - \mu_{jk})^2}{2\sigma_{jk}^2}\right), \quad (2.27)$$

em que μ_{jk} é a média da característica x_j para a classe C_k e σ_{jk}^2 é a variância da característica x_j para a classe C_k .

2.4 Aprendizado de Comitês

Aprendizado de Comitês (*Ensemble learning*) é uma abordagem na área de aprendizado de máquina onde múltiplos modelos de predição, chamados de *componentes* ou *classificadores-base*, são combinados para melhorar o desempenho geral e a capacidade de generalização do sistema (TAN et al., 2019, p. 226). Em vez de confiar em um único modelo para tomar decisões, o *ensemble learning* utiliza a combinação das predições de vários modelos para criar uma predição final mais robusta e precisa.

A ideia central por trás desta abordagem é a combinação de diferentes modelos, que podem ter forças e fraquezas distintas em diferentes áreas dos dados, pode levar a resultados mais confiáveis e estáveis do que o uso de um único modelo. Isso é especialmente útil quando há variação nos dados, ruído ou incerteza no processo de aprendizado.

2.4.1 Combinação de componentes (Majority Voting)

No processo de predição do comitê de um certo padrão os classificadores-base emitem suas predições e o rótulo que ocorrer com maior frequência é inferido como rótulo desta predição. As predições dos classificadores-base são combinadas usando médias ou médias ponderadas para chegar à predição do comitê.

2.4.1.1 Voto Majoritário Simples (Hard Voting)

O Voto Majoritário Simples é uma estratégia do *ensemble learning* mais conhecido, onde o objetivo é combinar as predições de diversos classificadores-base para chegar a uma decisão final. No contexto de votação rígida, cada componente emite uma predição individual para uma determinada entrada. A classe que obtém o maior número de votos entre todos os componentes é selecionada como a predição final, ou seja, o rótulo do comitê. (BALLABIO; TODESCHINI; CONSONNI, 2019).

Para ilustrar o processo, consideremos $y_{k,l}$ como a k -ésima saída do l -ésimo componente, e $y_{k,l}^*$ como a predição quantizada para 0 ou 1. Dessa forma, podemos representar o vetor de predições quantizadas do l -ésimo componente-base para um determinado padrão de entrada $\mathbf{x}$ como:

$$y_l^*(\mathbf{x}) = [y_{1,l}^*(\mathbf{x}), y_{2,l}^*(\mathbf{x}), \dots, y_{K,l}^*(\mathbf{x})]^T, \quad (2.28)$$

onde $y_l^*(\mathbf{x})$ é o vetor de predições quantizadas do l -ésimo componente para o padrão de entrada $\mathbf{x}$, e K é o número total de classes possíveis.

A soma dos vetores de predições $y_l^*(\mathbf{x})$ de todos os componentes resulta no vetor $\mathbf{v}(\mathbf{x})$, que representa a contribuição acumulada de cada componente na decisão final para o padrão de entrada $\mathbf{x}$. A decisão final sobre a classe mais votada pode ser inferida ao

aplicar a regra de decisão:

$$C(\mathbf{x}) = \arg \max_{k=1,\dots,m} \{\nu_k(\mathbf{x})\}, \quad (2.29)$$

onde $C(\mathbf{x})$ é o índice da classe mais votada para o padrão de entrada $\mathbf{x}$, k varia de 1 a m (número total de classes possíveis), e $\nu_k(\mathbf{x})$ é o k -ésimo elemento do vetor $\mathbf{v}(\mathbf{x})$. Essa estratégia busca a classe com a maior saída $\nu_k(\mathbf{x})$ entre todas as classes possíveis, determinando assim a classe final com o maior número de votos (DOGAN; BIRANT, 2019).

2.4.1.2 Votação Suave (Soft Voting)

A abordagem de votação por média simples busca obter uma saída única com base nas diversas saídas dos componentes. Nesse método, os resultados das previsões individuais não são quantizados e, em seguida, o padrão de entrada é atribuído à classe com o maior valor médio. Seja $y_{k,l}$ a k -ésima saída do l -ésimo componente (ou seja, saída não-quantizada). O vetor de saídas do l -ésimo componente do comitê para um dado padrão de entrada $\mathbf{x}$ é representado como:

$$\mathbf{y}_l(\mathbf{x}) = [y_{1,l}(\mathbf{x}), y_{2,l}(\mathbf{x}), \dots, y_{m,l}(\mathbf{x})]^T$$

em que m é o número de classes.

A média simples dos vetores de saída quantizada $\mathbf{y}_l^*(\mathbf{x})$ de todos os componentes-base de um comitê, para um dado padrão de entrada x , é definida como:

$$\mathbf{v}(\mathbf{x}) = \frac{1}{L} \sum_{l=1}^L \mathbf{y}_l(\mathbf{x})$$

em que L é o número total de classificadores no comitê (TAN et al., 2019).

2.4.1.3 Votação Suave Ponderada (Soft Voting Average)

Similar à média simples, a média ponderada calcula a média levando em consideração pesos atribuídos aos classificadores-base, em que cada peso w_i corresponde ao i -ésimo componente-base (OSAMOR; OKEZIE, 2021). A média ponderada dos vetores de saída quantizada $\mathbf{y}_l^*(\mathbf{x})$ de todos os componentes-base de um comitê é definida como:

$$\mathbf{v}(\mathbf{x}) = \sum_{l=1}^L w_l \mathbf{y}_l(\mathbf{x})$$

Em ambas as abordagens, após calcular o vetor $\mathbf{v}(\mathbf{x})$, é possível inferir a classe $C(\mathbf{x})$ mais votada do comitê através da seguinte regra de decisão:

$$C(\mathbf{x}) = \arg \max_{k=1,\dots,m} \{v_k(\mathbf{x})\}$$

onde $C(x)$ é o índice da classe mais votada para o padrão de entrada $\mathbf{x}$, k varia de 1 a m (número total de classes possíveis), e $v_k(\mathbf{x})$ é o k -ésimo elemento do vetor $\mathbf{v}(\mathbf{x})$.

2.4.2 Boosting

A técnica de *boosting* é um procedimento iterativo que visa melhorar o desempenho de classificadores base ao adaptativamente ajustar os pesos dos exemplos de treinamento ao longo das iterações (TAN et al., 2019). Inicialmente, cada exemplo recebe um peso igual, mas esses pesos são ajustados após cada rodada de treinamento, de modo que exemplos classificados incorretamente recebam maior peso, e os corretamente classificados tenham seus pesos reduzidos (FREUND; SCHAPIRE; ABE, 1999). Isso faz com que o modelo se concentre nos exemplos mais difíceis de classificar.

Matematicamente, a distribuição de amostragem utilizada no *boosting* pode ser representada por uma distribuição de probabilidade discreta sobre os exemplos de treinamento. A probabilidade $P_t(i)$ de selecionar o exemplo i na t -ésima iteração é dada por:

$$P_t(i) = \frac{w_i^t}{\sum_{j=1}^N w_j^t},$$

em que w_i^t é o peso do exemplo i na t -ésima iteração, e N é o número total de exemplos de treinamento. Essa distribuição guia a amostragem dos exemplos para a criação de subconjuntos de *bootstrap* ou para o treinamento direcionado dos classificadores base.

Ao longo dos anos, diversas implementações do *boosting* foram desenvolvidas, diferenciando-se principalmente na forma como atualizam os pesos dos exemplos de treinamento e como combinam as previsões dos classificadores gerados (FERREIRA; FIGUEIREDO, 2012).

2.4.2.1 Extreme Gradient Boosting (XGBoost)

O algoritmo *XGBoost* é um conjunto baseado em árvores de decisão que utiliza o arcabouço de *gradient boosting*. Trata-se de um algoritmo escalável de alto desempenho usado em aplicações de classificação e regressão, apresentando uma função de perda que inclui um termo de regularização que evita o sobre-ajuste (*overfitting*) (CHEN; GUESTRIN, 2016a).

O XGBoost apresenta diversas vantagens que o tornam uma escolha destacada em tarefas de aprendizado de máquina. Uma de suas principais vantagens é a menor necessidade de engenharia de características, como normalização e escalonamento de dados, devido à sua capacidade de lidar eficazmente com essas questões. Além disso, o XGBoost é robusto no tratamento de valores ausentes e permite a identificação da importância das características, o que facilita a interpretação do modelo e a seleção eficiente de atributos relevantes. Sua rapidez e eficiência superam a maioria dos algoritmos de aprendizado de máquina, tornando-o especialmente adequado para conjuntos de dados extensos e para a mitigação de problemas de sobre-ajuste (*overfitting*) (MIENYE; SUN, 2022b). O XGBoost

frequentemente supera outros algoritmos em termos de desempenho, o que o levou a conquistar inúmeras competições no Kaggle. No entanto, algumas limitações devem ser consideradas, como o elevado número de hiperparâmetros, que pode tornar sua calibração um desafio (ZHANG; ZHANG; JIANG, 2022).

2.4.2.2 Extreme Gradient Boosting (XGBoost)

O XGBoost (Extreme Gradient Boosting) é um algoritmo de aprendizado de máquina baseado em árvores de decisão que utiliza o arcabouço de *boosting* por gradiente e tem se destacado na área de aprendizado de máquina devido à sua escalabilidade e alta precisão (CHEN; GUESTRIN, 2016b), tornando-se uma escolha popular para aplicações de classificação e regressão. Este algoritmo se diferencia dos métodos convencionais de *boosting* por gradiente por meio de várias inovações, incluindo a introdução de um termo de regularização em sua equação a seguir:

$$L_M(F(x_i)) = \sum_{i=1}^n L(y_i, F(x_i)) + \sum_{m=1}^M \Omega(h_m) \quad (2.30)$$

em que $F(x_i)$ representa a previsão na i -ésima instância na M -ésima iteração, e $L(*)$ é a função de perda que calcula as diferenças entre a classe prevista e a classe real da variável alvo. O termo de regularização, $\Omega(h_m)$, é incluído para prevenir o sobreajuste e é formulado a seguir:

$$\Omega(h) = \gamma T + \frac{1}{2} \lambda \|\omega\|^2 \quad (2.31)$$

Nesta Equação 2.31, γ é o parâmetro de complexidade que controla a redução mínima de perda necessária para dividir um nó interno, T representa o número de folhas na árvore, λ é um parâmetro de penalização, e ω denota a saída dos nós folha. A introdução do termo de regularização permite que o XGBoost evite o overfitting (CHEN; GUESTRIN, 2016b), um problema comum em algoritmos de aprendizado de máquina.

Além disso, o XGBoost emprega uma aproximação de Taylor² de segunda ordem da função objetivo, conforme a Equação 2.32. Essa abordagem melhora a eficiência do algoritmo ao transformar a função objetivo em uma aproximação quadrática, o que otimiza o processo de treinamento (CHEN; GUESTRIN, 2016b):

$$L_M \approx \sum_{i=1}^n \left[g_i f_m(x_i) + \frac{1}{2} h_i f_m^2(x_i) \right] + \Omega(h_m) \quad (2.32)$$

aqui, g_i e h_i representam as primeiras e segundas derivadas da função de perda, respectivamente. A função objetivo final, calculada somando os valores de perda dos vários nós folha, é dada pela Equação 2.33:

² Técnica matemática que expande uma função não linear em uma série de termos que inclui a função em um ponto específico, sua primeira derivada (gradiente) e sua segunda derivada (matriz Hessiana).

$$L_M = \sum_{j=1}^T \left[\left(\sum_{i \in I_j} g_i \right) \omega_j + \frac{1}{2} \left(\sum_{i \in I_j} h_i + \lambda \right) \omega_j^2 \right] + \gamma T \quad (2.33)$$

Uma das principais vantagens do XGBoost é sua capacidade de lidar com grandes conjuntos de dados e valores ausentes, além de fornecer informações sobre a importância das características. Isso facilita a compreensão das variáveis de entrada e a realização de seleção de características. O XGBoost também se destaca por sua velocidade em comparação com muitos algoritmos de aprendizado de máquina, o que contribui para seu desempenho superior em competições, como as realizadas no Kaggle (MIENYE; SUN, 2022b).

No entanto, o XGBoost não é isento de limitações. A necessidade de ajustar um grande número de hiperparâmetros pode tornar sua sintonização complexa (ZHANG; ZHANG; JIANG, 2022). Apesar disso, o algoritmo continua a ser uma ferramenta valiosa no arsenal de métodos de aprendizado de máquina, oferecendo um desempenho robusto e eficiente em diversas aplicações.

2.4.3 Floresta Aleatória (Random Forest -RF)

O *Random Forest* é um método de aprendizado de máquina baseado em comitês que combina múltiplas árvores de decisão para melhorar a precisão preditiva e reduzir o sobreajuste (BREIMAN, 2001; CAIE; DIMITRIOU; ARANDJELOVIĆ, 2021). Esta técnica é eficaz para tarefas de classificação e regressão, sendo amplamente utilizada devido à sua robustez e capacidade de lidar com dados complexos e de alta dimensionalidade (TAN et al., 2019, p. 290).

O Random Forest constrói um conjunto (comitê) de árvores de decisão (BREIMAN, 2001), onde cada árvore é treinada com uma amostra diferente do conjunto de dados original, obtida por meio de um processo de *bootstrap* (amostragem com reposição). Em cada nó da árvore, uma divisão é feita utilizando um subconjunto aleatório de atributos, adicionando uma camada de aleatoriedade que melhora a diversidade e a performance geral do modelo (SUTHAHARAN, 2016).

2.4.3.1 Construção das Árvores de Decisão:

Cada árvore T_k é treinada com um subconjunto D_k do conjunto de dados D , selecionado por amostragem com reposição. Para um dado exemplo $\mathbf{x}$, a árvore gera uma predição $\hat{y}_k(\mathbf{x})$.

2.4.3.2 Votação Majoritária:

No caso de classificação, a predição final $\hat{y}(\mathbf{x})$ do Random Forest é obtida por votação majoritária entre as predições individuais das árvores:

$$\hat{y}(\mathbf{x}) = \text{mode}(\{\hat{y}_k(\mathbf{x}) : k = 1, 2, \dots, N\}) \quad (2.34)$$

em que N é o número total de árvores no comitê (BREIMAN, 2001).

2.4.3.3 Divisões Baseadas no Índice de Gini:

Cada árvore realiza divisões nos nós baseando-se em medidas de impureza, como o índice de Gini. A impureza de um nó para um conjunto de amostras S é dada por:

$$IG(S) = 1 - \sum_{j=1}^c P_j^2 \quad (2.35)$$

em que c é o número total de classes e P_i é a proporção de amostras em S pertencentes à classe C_j (ALGEHYNE et al., 2022).

Em suma, o Random Forest utiliza um comitê de árvores de decisão para fornecer predições robustas e precisas, beneficiando-se da combinação de múltiplas árvores para reduzir o risco de sobre-ajuste e aumentar a capacidade de generalização do modelo.

2.5 Estimação dos Parâmetros

Para determinar os parâmetros μ_{jk} e σ_{jk}^2 que definem a distribuição Gaussiana de cada característica para cada classe, utilizamos o método da máxima verossimilhança. Suponha que temos um conjunto de dados $\mathbf{x} = (x_1, \dots, x_N)^T$ representando N observações da variável x , que são independentes e identicamente distribuídas (i.i.d.) segundo uma Gaussiana com média μ e variância σ^2 (BISHOP, 2016, pp. 26-27).

A função de verossimilhança dos dados, dada a média μ e a variância σ^2 , é dada por:

$$p(\mathbf{x}|\mu, \sigma^2) = \prod_{n=1}^N \mathcal{N}(x_n|\mu, \sigma^2), \quad (2.36)$$

em que $\mathcal{N}(x_n|\mu, \sigma^2)$ é a função densidade da Gaussiana para a n -ésima observação. Para simplificar a maximização, considera-se a função log-verossimilhança, dada por:

$$\ln p(\mathbf{x}|\mu, \sigma^2) = -\frac{1}{2\sigma^2} \sum_{n=1}^N (x_n - \mu)^2 - \frac{N}{2} \ln(2\pi\sigma^2). \quad (2.37)$$

Maximizando a log-verossimilhança em relação à μ e σ^2 , obtemos as estimativas:

$$\hat{\mu} = \frac{1}{N} \sum_{n=1}^N x_n, \quad (2.38)$$

$$\hat{\sigma}^2 = \frac{1}{N} \sum_{n=1}^N (x_n - \hat{\mu})^2. \quad (2.39)$$

Essas equações fornecem as estimativas de máxima verossimilhança para a média e variância da Gaussiana, usadas no Naive Bayes Gaussiano para modelar as distribuições das características dentro de cada classe.

Para classificar uma nova amostra $\mathbf{x}$, o Naive Bayes Gaussiano calcula a probabilidade posterior para cada classe usando a Equação 2.25 e escolhe a classe com a maior probabilidade posterior.

2.6 Estimativa de confiança

A estimativa de confiança é o processo de atribuição de uma pontuação de confiança à predição de um modelo. Essa pontuação pode ser usada para determinar o quão provável é que o modelo esteja correto. Existem muitos métodos diferentes para estimativa de confiança, e o melhor método a ser usado depende do tipo de modelo e da aplicação.

2.6.1 Regra de Decisão de Bayes

A regra de decisão de Bayes é um método para tomar decisões em situações de incerteza, baseado na ideia de que a melhor decisão é aquela que maximiza a probabilidade de ser correta. Essa regra é expressa pela equação (FUKUNAGA, 1990):

$$P(C_i|\mathbf{x}) = \frac{p(C_i) \cdot p(\mathbf{x}|C_i)}{\sum_j p(C_j) \cdot p(\mathbf{x}|C_j)},$$

em que:

- $\mathbf{x}$ é o vetor de características observado,
- C_i é uma possível classe,
- $p(C_i|\mathbf{x})$ é a probabilidade a posteriori de C_i dado $\mathbf{x}$,
- $p(C_i)$ é a probabilidade a priori da classe C_i ,
- $p(\mathbf{x}|C_i)$ é a probabilidade da observação $\mathbf{x}$ dado que a classe é C_i ,
- $\sum_j p(C_j) \cdot p(\mathbf{x}|C_j)$ é a probabilidade total da observação $\mathbf{x}$ considerando todas as classes possíveis.

A regra de decisão de Bayes escolhe a classe C_i que maximiza a probabilidade a posteriori $P(C_i|\mathbf{x})$.

2.6.2 Teoria de Decisão Bayesiana e Critério Máximo a Posteriori (MAP)

A *Teoria de Decisão Bayesiana* fornece uma estrutura probabilística para a tomada de decisões em condições de incerteza. É amplamente utilizada em problemas de classificação, onde o objetivo é, dado um conjunto de observações, identificar a classe ou hipótese mais provável. Uma das abordagens centrais dentro desta teoria é o *Critério do Máximo a Posteriori (MAP)* (CHOW, 1970), que busca selecionar a classe que maximiza a probabilidade a posteriori de uma observação.

2.6.2.1 Definição Formal

Seja $C = \{c_1, c_2, \dots, c_k\}$ o conjunto de classes e x uma observação, o critério MAP seleciona a classe c^* que maximiza a probabilidade a posteriori $P(c_i|x)$, conforme descrito pela Equação 2.40:

$$c^* = \arg \max_{c_i} P(c_i|\mathbf{x}) \quad (2.40)$$

A probabilidade a posteriori $P(c_i|x)$ pode ser obtida utilizando o *Teorema de Bayes*, que relaciona as probabilidades condicionais como segue:

$$P(c_i|\mathbf{x}) = \frac{P(\mathbf{x}|c_i)P(c_i)}{P(\mathbf{x})} \quad (2.41)$$

onde:

- $P(\mathbf{x}|c_i)$ é a *verossimilhança*, ou seja, a probabilidade de observar $\mathbf{x}$ dado que a classe é c_i ;
- $P(c_i)$ é a *probabilidade a priori* da classe c_i , que representa o conhecimento prévio sobre essa classe antes de observar $\mathbf{x}$;
- $P(\mathbf{x})$ é a *evidência*, que corresponde à probabilidade de $\mathbf{x}$ ser observada considerando todas as classes possíveis.

Como o denominador $P(\mathbf{x})$ é constante para todas as classes, pode ser ignorado no processo de maximização. Portanto, o critério MAP pode ser simplificado para a maximização do produto da verossimilhança e da probabilidade a priori, como apresentado na Equação 2.42:

$$c^* = \arg \max_{c_i} P(\mathbf{x}|c_i)P(c_i) \quad (2.42)$$

A confiança é determinada pela maior probabilidade a posteriori entre todas as classes C_i , refletindo o grau de certeza da decisão do classificador. Uma maior probabilidade a posteriori máxima implica em maior confiança na decisão do classificador.

Embora a MAP seja uma estimativa simples de confiança, ela pode ser sensível a *outliers* (KHATIBISEPEHR; HUANG, 2012).

O critério MAP seleciona a classe c^* que maximiza o produto da verossimilhança $P(\mathbf{x}|c_i)$ e da probabilidade a priori $P(c_i)$. A verossimilhança reflete o quanto a observação $\mathbf{x}$ é consistente com a classe c_i , enquanto a probabilidade a priori incorpora o conhecimento anterior sobre a distribuição das classes (LEE, 2012). Assim, o MAP equilibra a evidência fornecida pelos dados observados e o conhecimento prévio disponível, permitindo decisões mais robustas em cenários onde o conhecimento a priori é confiável.

Esse critério pode ser contrastado com o *Critério de Máxima Verossimilhança* (*Maximum Likelihood* - ML), que considera apenas a verossimilhança $P(\mathbf{x}|c_i)$, desconsiderando a priori $P(c_i)$. Dessa forma, o MAP pode ser visto como uma generalização do ML, sendo mais robusto em situações onde há incerteza ou quando se dispõe de informações prévias sobre as classes (HAYES, 2017).

2.6.3 Probabilidades a Priori das Classes

Outra forma de estimar a confiança na regra de decisão de Bayes é utilizando as probabilidades a priori das classes. A confiança pode ser calculada como o produto da probabilidade a priori da classe e a densidade condicional de classe, conforme a equação:

$$\text{Confiança}(C_i|\mathbf{x}) = p(C_i) \cdot p(\mathbf{x}|C_i),$$

em que:

- $p(C_i)$ é a probabilidade a priori da classe C_i ,
- $p(\mathbf{x}|C_i)$ é a densidade condicional da observação $\mathbf{x}$ dado C_i .

Este método é mais robusto a *outliers* comparado à MAP, mas pode ser menos preciso para entradas ambíguas (DUBUISSON; MASSON, 1993).

2.6.4 Regra do vizinho mais próximo

A regra do vizinho mais próximo é um método de classificação não paramétrico que é simples de implementar e eficiente de computar. Ele funciona procurando os k vizinhos mais próximos da amostra de teste no conjunto de treinamento e atribuindo a amostra de teste à classe que é a mais comum entre seus k vizinhos (WEINBERGER; SAUL, 2009). Esta não fornece uma estimativa de confiança explícita para sua predição. No entanto, é possível estimar a confiança de uma predição da regra do vizinho mais próximo usando um dos seguintes métodos (HELLMAN, 1970):

- Proporção de votos: A proporção de votos é a proporção de k vizinhos da amostra de teste que pertencem à classe predita. Uma proporção de votos alta indica

uma maior confiança na predição.

- Distância mínima: A distância mínima é a distância da amostra de teste ao seu vizinho mais próximo da classe predita. Uma distância mínima baixa indica uma maior confiança na predição.
- Filtragem de outliers: A filtragem de outliers é um método para remover amostras de treinamento que são outliers de suas classes, possibilitando melhoria na confiança das predições da regra do vizinho mais próximo.

2.6.5 Platt Scaling

Platt Scaling é uma técnica de calibração de probabilidades que transforma as saídas de classificadores em estimativas de probabilidade. Originalmente proposta por Platt (PLATT, 2000), essa técnica foi desenvolvida para máquinas de vetores de suporte (SVM), mas pode ser aplicada a outros modelos de classificação (PLATT, 2000). O método ajusta uma função sigmoide às saídas do classificador, convertendo os scores de decisão em probabilidades calibradas.

A equação da sigmoide usada no Platt Scaling é dada por:

$$p(x) = \frac{1}{1 + \exp(Af(x) + B)} \quad (2.43)$$

onde A e B são parâmetros ajustados minimizando a função de custo de log-verossimilhança negativa (Negative Log-Likelihood, NLL) dos dados de treinamento (PLATT, 2000). Essa calibração possibilita uma interpretação probabilística dos scores de saída, tornando as predições mais interpretáveis e adequadas para aplicações que requerem estimativas de confiança.

2.6.6 Probabilidade Máxima de Classe (MCP)

Probabilidade Máxima de Classe (Maximum Class Probability - MCP) é uma abordagem simples para estimar a confiança em classificadores, particularmente em redes neurais (HENDRYCKS; GIMPEL, 2018). Após aplicar uma função *softmax* na camada final da rede, as saídas dos neurônios representam as probabilidades de pertencimento às diferentes classes. A MCP utiliza a maior probabilidade entre todas as classes como medida de confiança para a previsão.

Sejam as probabilidades das classes representadas por $\{P_1, P_2, \dots, P_c\}$, onde $P_i \in [0, 1]$ e $\sum_{i=1}^c P_i = 1$. Definimos:

- $\kappa_1 = \arg \max_i P_i$: índice da classe com a maior probabilidade.
- P_{κ_1} : probabilidade correspondente à classe κ_1 .

A confiança é dada diretamente por:

$$\text{MCP} = P_{\kappa_1} \quad (2.44)$$

Além disso, uma variação comum é calcular a diferença entre as probabilidades da classe de maior valor e da segunda maior probabilidade, $P_{\kappa_1} - P_{\kappa_2}$, para melhorar a distinção em cenários onde as classes têm probabilidades próximas (CORDELLA et al., 1995). Essa técnica é considerada um baseline robusto para a detecção de erros de classificação e amostras fora da distribuição, sendo amplamente utilizada por sua simplicidade e eficácia (HENDRYCKS; GIMPEL, 2018).

2.6.7 Índice de Impuridade de Gini

O índice de Gini é uma medida de impuridade (ou heterogeneidade) que representa a desigualdade de uma distribuição (YUAN; WU; ZHANG, 2021). No contexto deste trabalho, o *índice de Gini* é uma medida de impureza ou heterogeneidade que avalia a distribuição das probabilidades atribuídas a diferentes classes por um classificador probabilístico, refletindo impureza da distribuição das probabilidades das classes (ROCHA NETO, 2011, p. 97).

2.6.8 Definição do Índice de Gini

Para um dado conjunto de amostras, o índice de Gini IG é definido como:

$$IG = 1 - \sum_{j=1}^n P(C_j|\mathbf{x}_i)^2 \quad (2.45)$$

em que n é o número total de classes e $P(C_j|\mathbf{x}_i)$ é a probabilidade da j -ésima classe C_j para o i -ésimo exemplo $\mathbf{x}_i$.

Considere um conjunto de probabilidades de classes para uma amostra, em que as probabilidades das classes são $P(C_1|\mathbf{x}) = 0,6$, $P(C_2|\mathbf{x}) = 0,3$, e $P(C_3|\mathbf{x}) = 0,1$. O índice de Gini para este conjunto pode ser calculado como:

$$IG = 1 - (0,6^2 + 0,3^2 + 0,1^2) = 1 - (0,36 + 0,09 + 0,01) = 1 - 0,46 = 0,54.$$

Para o índice de Gini, um valor de 0 indica que todos os rótulos dos padrões são da mesma classe, enquanto um valor próximo de 1 indica que todos os rótulos são diferentes. O limiar de impureza (de rejeição) também é definido em um processo de otimização que busca minimizar a equação de risco empírico.

2.7 Conclusão

Esta seção abordou os conceitos fundamentais dos algoritmos de aprendizado supervisionado, semi-supervisionado e não supervisionado, fornecendo uma base abrangente para compreender suas distinções e aplicações. Discutimos algoritmos tradicionais, como MLP, k-NN, SVM e GNB, destacando suas capacidades e limitações em problemas de múltiplas classes. Além disso, exploramos modelos de comitês, como Votação, Boosting (com foco no XGBoost) e Random Forest (RF), enfatizando suas vantagens em melhorar o desempenho preditivo por meio da combinação de múltiplos modelos. Também introduzimos o conceito de estimativas de confiança, que são importantes para avaliar a robustez das predições dos classificadores. Por fim, discutimos o uso do Índice de Gini como uma métrica para avaliar a qualidade dos modelos, especialmente no contexto de classificadores probabilísticos, destacando sua utilidade em medir a impureza ou heterogeneidade das predições. Essa análise fornece uma melhor compreensão para o desenvolvimento das abordagens em classificação e avaliação de modelos usados neste trabalho.

3 Classificação com opção de rejeição

Este capítulo está organizado em sete seções. A seção 3.1 fornece uma definição teórica da classificação com opção de rejeição. A estratégia de rejeição, que consiste em descartar observações menos prováveis, é explicada na seção 3.2, onde é apresentada a regra de Chow. Os tipos de abordagens de classificação com opção de rejeição são explicadas na seção 3.3. As Seções 3.4 e 3.5 detalham a implementação das duas abordagens empregadas neste estudo. A representação gráfica utilizada para avaliar o desempenho dessa técnica é explicada na Seção 3.6. Por fim, as principais conclusões deste capítulo são resumidas na Seção 3.7.

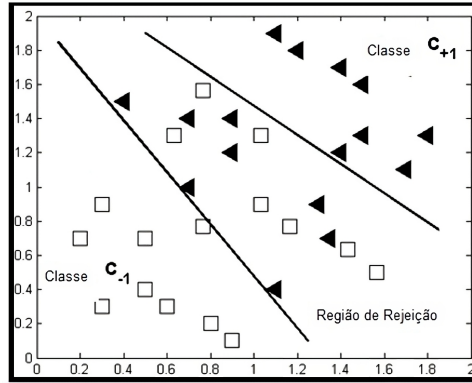
A classificação com opção de rejeição, ou aprendizado de rejeição (HENDRICKX et al., 2021), é uma abordagem de aprendizado de máquina que visa melhorar a precisão dos modelos, permitindo que eles se abstenham de classificar exemplos considerados difíceis ou ambíguos com baixa confiança. Essa estratégia categoriza padrões com alta probabilidade de erro, que podem ser rejeitados para posterior análise por métodos alternativos ou especialistas humanos, aumentando assim a chance de classificações corretas entre os padrões não rejeitados. Ao evitar decisões precipitadas em casos de incerteza, essa abordagem promove maior confiabilidade e precisão na tomada de decisões.

3.1 Definição Teórica

Uma tarefa de classificação para um problema binário é normalmente definido com rótulo igual a -1 ou $+1$ (ou alternativamente com rótulo igual a 0 ou 1). A tomada de decisão é feita com base em estimativas de probabilidades a posteriori $p(C_j|\mathbf{x})$, para um dado vetor de entrada $\mathbf{x}$, mesmo quando a diferença entre os valores das probabilidades a posteriori das classes é pequena. Isso resulta em uma automatização das decisões mais difíceis, podendo conduzir a predições erradas e, portanto, à elevação do erro de classificação (ROCHA NETO, 2011).

Como dito, é comum em problemas binários rotular os vetores de entrada $\mathbf{x}$ com valor igual a (-1) ou $(+1)$. Porém, é interessante em muitos ambientes que tais sistemas rejeitem casos críticos, evitando maior confusão e delegando-os assim para a avaliação de um especialista. Nesse sentido, uma nova classe pode ser definida, chamada de classe de rejeição, tal que seus padrões dispõem-se na região entre as classes (ROCHA NETO et al., 2011). A Figura 1 exemplifica a classificação binária com opção de rejeição.

Figura 1 – Exemplo de classificação binária com opção de rejeição.



Fonte: Sousa, Mora e Cardoso (2009)

A métrica principal para avaliar o desempenho em problemas de classificação de padrões é a taxa de erro do classificador, que mede as decisões equivocadas em relação ao total de decisões. Menos usada, mas relevante, é a taxa de rejeição, indicando decisões descartadas em relação ao total (ROCHA NETO, 2011).

Erro ocorre quando um padrão é classificado equivocadamente, enquanto a rejeição ocorre quando o sistema não decide. O padrão rejeitado é então repassado e analisado por especialistas ou outros classificadores. No entanto, o equilíbrio entre a taxa de erro e de rejeição é crucial (CHOW, 1970).

Erros são inerentes à tarefa de classificação de padrões. A opção de rejeição evita erros em decisões complexas, transformando-os em rejeições. Porém, pode converter decisões corretas em rejeições. Devido aos custos variados, é necessário avaliar o desempenho com diferentes custos de rejeição para escolher o melhor classificador conforme o problema.

3.2 Regra de decisão ótima

Em problemas de classificação com opção de rejeição, a regra de decisão ótima visa a minimizar a taxa de erro considerando uma taxa de rejeição específica, determinada pelo custo de rejeição w_r . A regra determina a recusa de um padrão quando a maior probabilidade a posteriori é inferior a um limiar t (CHOW, 1957).

Mais explicitamente, a regra de decisão ótima $\delta(d_k|\mathbf{x})$ é dada por:

$$(a) \quad \delta(d_k | \mathbf{x}) = 1, \quad (k \neq 0), \quad (3.1)$$

$$(b) \quad \delta(d_0 | \mathbf{x}) = 1, \quad (k = 0). \quad (3.2)$$

em que d_0 representa a decisão de rejeitar o padrão $\mathbf{x}$ e d_k representa a decisão de aceitar o padrão $\mathbf{x}$ e atribuí-lo à classe $k = 1, \dots, c$, tal que c representa o número de classes do problema. A Equação (3.1) considera que se aceite o padrão $\mathbf{x}$ para fins de reconhecimento

e identifique-o como pertencente à k -ésima classe, se

$$p_k F(\mathbf{x}|C_k) \geq p_j F(\mathbf{x}|C_j) \quad \forall j = 1, 2, \dots, n, \quad (3.3)$$

e

$$p_k F(\mathbf{x}|C_k) \geq (1-t) \sum_{i=1}^c p_i F(\mathbf{x}|C_i). \quad (3.4)$$

A Equação 3.2, considera que se rejeite o padrão, se

$$\max_i \{p_i F(\mathbf{x}|C_i)\} < (1-t) \sum_{i=1}^c p_i F(\mathbf{x}|C_i). \quad (3.5)$$

onde $\mathbf{x}$ é o vetor padrão, c é o número de classes, $(P_1, P_2, \dots, P_c)$ é a distribuição de probabilidade a priori das classes, $F(\mathbf{x}|i)$ é a densidade de probabilidade condicional para $\mathbf{x}$, dada a i -ésima classe, $d_i (i \neq 0)$ é a decisão de que $\mathbf{x}$ é identificado como da i -ésima classe enquanto d_0 é a decisão de rejeitar, e t é uma constante entre 0 e 1 ($0 \leq t \leq 1$) (ALMEIDA, 2021). A probabilidade de erro, ou taxa de erro, é dada por:

$$E(t) = \int_x \sum_{i=1}^c \sum_{\substack{x=1 \\ x \neq 1}}^c \delta(d_i|\mathbf{x}) p_i F(\mathbf{x}|C_i) dx, \quad (3.6)$$

e a probabilidade de taxa de rejeição, ou rejeição, é:

$$R(t) = \int_x \delta(d_i|\mathbf{x}) \sum_{i=1}^c p_i F(\mathbf{x}|C_i) dx. \quad (3.7)$$

Ambas as taxas de erro são funções implícitas do parâmetro t . A probabilidade de reconhecimento correto é:

$$C(t) = \int_x \sum_{i=1}^c \delta(d_i|\mathbf{x}) p_i F(\mathbf{x}|C_i) dx = 1 - E(t) - R(t), \quad (3.8)$$

e a probabilidade de aceitação (ou taxa de aceitação) é definida como:

$$A(t) = C(t) + E(t). \quad (3.9)$$

Definindo $m(x)$ como

$$m(\mathbf{x}) = \frac{p_i F(\mathbf{x}|C_i)}{\sum_{i=1}^c p_i F(\mathbf{x}|C_i)}. \quad (3.10)$$

Dessa forma, a regra ótima de reconhecimento pode ser simplificada em duas situações: aceitar o padrão $\mathbf{x}$ quando $m(\mathbf{x}) \geq (1-t)$ e rejeitar o padrão $\mathbf{x}$ quando $m(\mathbf{x}) < (1-t)$ (CHOW, 1957).

O parâmetro t na regra de decisão é chamado de limiar de rejeição. Para qualquer valor fixo de t (onde $0 \leq t \leq 1$), a regra de decisão δ particiona o espaço padrão V em

dois conjuntos (ou regiões) disjuntos $V_A(t)$ e $V_R(t)$, em que a equação (3.4) e a equação (3.5) correspondem a esses conjuntos, respectivamente.

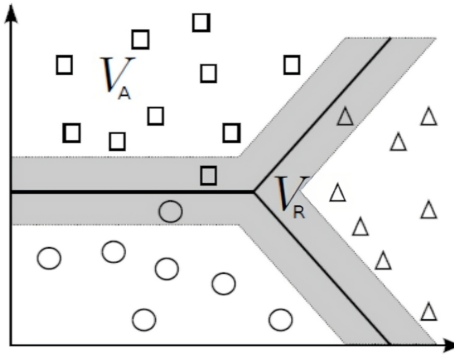
$$V_A(t) = \{\mathbf{x} | m(\mathbf{x}) \geq (1 - t)\} \quad (3.11)$$

e

$$V_R(t) = \{\mathbf{x} | m(\mathbf{x}) < (1 - t)\} \quad (3.12)$$

Sem perda de generalidade, será assumido que $F(x)$ é diferente de zero em todo o espaço V ; caso contrário, o conjunto sobre o qual $F(x)$ é zero é excluído primeiro. As regiões V_A e V_R são chamadas, respectivamente, de região de aceitação e região de rejeição da regra de decisão. A Figura 2 ilustra as regiões descritas pelos conjuntos disjuntos V_A e V_R , onde a região sombreada é V_R e a região não sombreada é V_A .

Figura 2 – Regiões de rejeição e aceitação da regra ótima de reconhecimento.

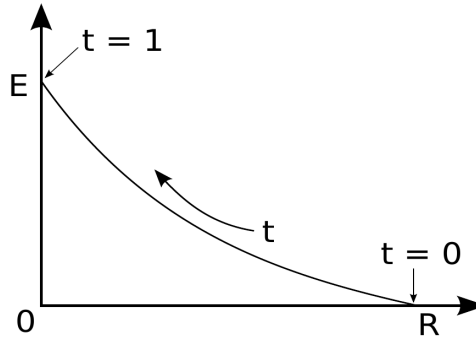


Fonte: ROCHA NETO 2011

No contexto do limiar de rejeição, Chow (1957) destaca duas propriedades essenciais: a monotonicidade das taxas de erro e rejeição em relação a t , e o caráter limitante superior de t na taxa de erro.

Uma descrição completa do desempenho de um sistema de reconhecimento é determinada pela relação de rejeição e erros, isto é, a relação funcional de erro (E) e rejeição (R) todos os níveis. A Figura 3 exemplifica essa relação, onde o aumento na rejeição reduz a taxa de erro, e vice-versa. Como a taxa de erro $E(t)$ e a taxa de rejeição $R(t)$ de um sistema ótimo são funções monótonas do limiar t , é possível quantificar essa relação para um determinado limiar (OLIVEIRA, 2016).

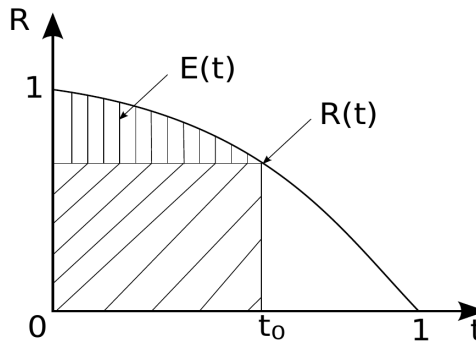
Figura 3 – Curva de compromisso entre o erro e a rejeição.



Fonte: ROCHA NETO 2011

O estudo de Chow indica que a taxa de rejeição descreve o desempenho do sistema de reconhecimento em relação ao limiar t . Na Figura 4, é ilustrado um exemplo típico desse perfil de curva, com as regiões delimitando a taxa de erro em situações sem rejeição (soma das áreas hachuradas) ou permitindo rejeições (região superior hachurada) (CHOW, 1957).

Figura 4 – Curva hipotética de compromisso entre o erro e a rejeição.



Fonte: ROCHA NETO 2011

Show (1957) também afirma que a regra de decisão ótima dada em (3.1) a (3.5) também é uma regra de risco mínimo se a função de custo for uniforme dentro de cada classe de decisões, ou seja, se nenhuma distinção for feita entre os erros, entre os rejeitados e o reconhecimento correto. O limite de rejeição está relacionado aos custos da seguinte maneira

$$t = \frac{w_r - w_c}{w_e - w_c} \quad (3.13)$$

Nesse contexto, w_r , w_e e w_c representam os custos de rejeição, erro e acerto, respectivamente. A regra de decisão ótima, expressa nas Equações (3.1) a (3.5), é também uma regra de risco mínimo quando os custos associados aos erros, acertos e rejeições são iguais. Geralmente, $w_c < w_r < w_e$. Considerando $w_c = 0$ e $w_e = 1$, podemos inferir que o custo de rejeição w_r é igual ao limiar t ($w_r = t$). Sendo assim, o risco mínimo para um

dado limiar t é dado por:

$$r(t) = E(t) + tR(t) = \int_0^t R(t)dt \quad (3.14)$$

o que corresponde à área hachurada na Figura 4. Enquanto o risco empírico, para uma função de decisão e um determinado conjunto de treinamento, é definido como:

$$r_{emp}(\alpha) = E(\alpha) + w_r R(\alpha) \quad (3.15)$$

Aqui, α representa o vetor de parâmetros do modelo, incluindo o limiar de rejeição. Em problemas de classificação binária, a condição de rejeição descrita na Equação (3.5) pode não ser atendida quando $t > 0.5$, uma vez que a taxa de rejeição é sempre nula se t exceder 0.5. Isso justifica por que w_r está sempre limitado ao intervalo $[0, 0.5]$.

3.3 Abordagens de Classificação Binária com Opção de Rejeição

Na classificação binária, o espaço das características é dividido em duas regiões pelo classificador desenvolvido, onde cada região representa uma das classes. O classificador binário com opção de rejeição pode ser implementado através de três diferentes abordagens, a saber: (i) usando apenas um classificador; (ii) usando dois classificadores independentes; e (iii) usando um classificador com opção de rejeição incorporada na regra de aprendizado.

3.3.1 Um único classificador binário padrão

Um padrão é rejeitado se o máximo das duas probabilidades for menor que um determinado limiar. Se o classificador não fornecer saídas probabilísticas utiliza-se um limiar de rejeição. A região de rejeição é determinada após o treinamento do classificador, definindo valores limite adequados sobre a saída do classificador (ALMEIDA, 2010).

3.3.2 Dois Classificadores Independentes

Nessa abordagem são utilizados dois classificadores denotados por $\mathbf{m}_1$ e $\mathbf{m}_2$. O treinamento do classificador $\mathbf{m}_1$ é realizado de tal forma que penalize os erros do tipo falso negativo. Por outro lado, o classificador $\mathbf{m}_2$ é treinado com o intuito de penalizar os erros do tipo falso positivo. Desta forma, o classificador $\mathbf{m}_1$ gera elevado valor de probabilidade para exemplos da classe C_{+1} , enquanto o classificador $\mathbf{m}_2$ produz elevado valor para exemplos da classe C_{-1} . Um exemplo é rejeitado sempre que os classificadores discordarem (ALMEIDA, 2010; ROCHA NETO, 2011).

3.3.3 Classificador com Opção de Rejeição Incorporada

Esta abordagem consiste em projetar classificadores que trazem embutido no processo de otimização de suas funções-custo a capacidade de classificação com opção de

rejeição. Assim, não é necessário uma etapa para obtenção do limiar de rejeição ótimo, também não é necessário especializar-se uma determinada classe como ocorre na abordagem com dois classificadores independentes. Na abordagem com opção de rejeição embutida, o modelo aprende intrinsecamente a decidir entre as três saídas possíveis, C_{-1} , rejeição e C_{+1} (FUMERA; ROLI, 2002; BOUNSIAR; BEAUSEROY; GRALL-MAËS, 2008; SOUSA; MORA; CARDOSO, 2009).

3.3.4 Classificação Multi-Classe com Opção de Rejeição

A abordagem de classificação com opção de rejeição pode ser estendida para problemas multi-classe. Em tal abordagem, um classificador $\mathbf{m}$ fornece medidas de confiança para cada uma das classes C_j . A decisão de aceitar ou rejeitar um exemplo é baseada em um limiar t . Se as medidas de confiança, onde $\text{confiança} = \max_j C_j$, são maiores ou iguais a t , o exemplo é aceito. Caso contrário, é rejeitado. O objetivo é determinar um valor ótimo para t , de modo a rejeitar exemplos que são classificados erroneamente pelo classificador, enquanto se preserva a correta classificação dos exemplos (ALMEIDA, 2010; ALMEIDA, 2021).

3.3.5 Classificadores com opção de rejeição

Neste trabalho usamos os classificadores clássicos, como perceptron multicamadas (RO_MLP), máquinas de vetores de suporte (RO_SVM) e k-vizinhos mais próximos (RO_K-NN) e para os modelos em comitê floresta aleatória (RO_RF), *extreme gradient boosting* (RO_XGB) e votação (RO_VOTING), assim identificados para simbolizar a incorporação da função de classificação com opção de rejeição no processo de treinamento e predição dos modelos.

Para determinar a aceitação ou rejeição de um padrão observado, utilizamos o Índice de Gini como medida de confiança. O Índice de Gini, adaptado neste contexto (Eq. (2.45)), avalia a pureza da distribuição das probabilidades de classe associadas a uma amostra.

3.3.5.1 Regra de Decisão

A decisão de rejeitar ou classificar a amostra é feita comparando o Índice de Gini (IG) com um limiar t de rejeição, aqui identificado por w_r . A regra de decisão é formalmente expressa como:

- **Aceitar a Classificação:** Se $IG(\mathbf{x}) \leq t$, a amostra $\mathbf{x}_i$ é classificada na classe com a maior probabilidade, definida como:

$$C_i = \arg \max_i \{p_i\}$$

- **Rejeitar a Classificação:** Se $IG(\mathbf{x}) > t$, a amostra $\mathbf{x}$ é rejeitada, indicando que a decisão não é confiável o suficiente para proceder com a classificação.

Abaixo está um algoritmo simplificado que representa o funcionamento do classificador com opção de rejeição utilizando o Índice de Gini:

Algoritmo 1 Classificador com Opção de Rejeição

```

1: Procedimento RO_CLASSIFICADOR(probabilidades, t)
2:   gini  $\leftarrow$  IG(probabilidades)
3:   Se gini  $\leq$  t Então
4:     return  $\arg \max\{p_i\}$ 
5:   Senão
6:     return -1                                      $\triangleright$  Rejeita a classificação
7:   Fim Se

8: Exemplo de Uso
9: probabilidades  $\leftarrow$  [0.7, 0.2, 0.1]            $\triangleright$  Exemplo de probabilidades para 3 classes
10: t  $\leftarrow$  0.4                                      $\triangleright$  Limiar de rejeição
11: resultado  $\leftarrow$  RO_CLASSIFICADOR(probabilidades, t)
12: print resultado

```

O modelo RO_VOTING diferencia-se dos demais modelos por incorporar um passo adicional na sua regra de decisão. Quando enfrenta um empate entre um conjunto de classes ou rejeições, o modelo opta pela rejeição. Dessa forma, a definição da saída do modelo de votação com opção de rejeição é a seguinte:

Algoritmo 2 RO_VOTING com Opção de Rejeição

```

1: Procedimento RO_VOTING(probabilidades_comitê,  $w_r$ )
2:   decisoes  $\leftarrow []$  ▷ Lista de decisões dos classificadores base
3:   Para cada probabilidades em probabilidades_comitê faça
4:     gini  $\leftarrow$  IG(probabilidades)
5:     Se gini  $\leq w_r$  Então
6:       decisoes.append( $\arg \max\{p_i\}$ ) ▷ Classifica na classe com maior
       probabilidade
7:     Senão
8:       decisoes.append(-1) ▷ Rejeita a classificação
9:     Fim Se
10:  Fim Para
11:  vencedora  $\leftarrow$  VOTACAO(decisoes) ▷ Votação sobre as decisões dos classificadores
12:  return vencedora

13: Procedimento VOTACAO(decisoes)
14:  moda  $\leftarrow$  Moda simples de decisoes ▷ Identifica a classe com a maior frequência
15:  Se moda  $> 1$  Então ▷ Se existir mais de uma moda
16:    return -1 ▷ Rejeita
17:  Senão
18:    return moda ▷ Classifica
19:  Fim Se

20: Exemplo de Uso
21: probabilidades_comitê  $\leftarrow$  [[0.7, 0.2, 0.1], [0.6, 0.3, 0.1], [0.4, 0.4, 0.2]]
22:  $w_r \leftarrow 0.4$  ▷ Limiar de rejeição
23: resultado  $\leftarrow$  RO_VOTING(probabilidades_comitê,  $w_r$ )
24: print resultado

```

3.3.6 Avaliação de Classificador com Opção de Rejeição

A curva de Rejeição de Precisão (ARC) é uma ferramenta crucial na avaliação de classificadores, oferecendo uma abordagem visual e analítica para entender o equilíbrio entre precisão e taxa de rejeição. Essa curva é gerada plotando a precisão de um classificador em relação à sua taxa de rejeição, variando de 0 a 1. A curva ARC revela o comportamento dos classificadores em diferentes cenários de rejeição (ALMEIDA, 2021; OLIVEIRA, 2016).

Existem três tipos principais de relações entre curvas ARC:

1. Tipo 1 (Crossing-over): Uma curva ARC cruza outra, indicando que um classificador supera consistentemente o outro em várias taxas de rejeição.
2. Tipo 2 (Diverging): As curvas ARC inicialmente se sobrepõem em baixas taxas de rejeição, mas se afastam à medida que as taxas de rejeição aumentam, indicando que um classificador pode ser mais adequado em diferentes faixas de rejeição.

3. Tipo 3 (Evenly spaced): A distância entre as curvas ARC permanece uniforme em várias taxas de rejeição, destacando que os classificadores mantêm relações de desempenho constantes em diferentes cenários.

A curva ARC é valiosa para selecionar o classificador de melhor desempenho. Se a precisão desejada é conhecida, você pode escolher o classificador com a menor taxa de rejeição para alcançar essa precisão. Por outro lado, se a taxa de rejeição aceitável é conhecida, você pode selecionar o classificador com a maior precisão para essa taxa de rejeição.

3.4 Conclusão

Neste capítulo, abordou-se a classificação com opção de rejeição, uma técnica avançada em aprendizado de máquina que visa melhorar a precisão dos modelos ao permitir que eles se abstenham de classificar exemplos quando a confiança é baixa. Esta abordagem é particularmente útil para problemas em que a classificação errônea pode ter consequências significativas.

Na seção 3.1 introduziu a definição teórica da classificação com opção de rejeição, explicando como essa técnica permite a inclusão de uma classe adicional, a classe de rejeição, para lidar com casos críticos que não podem ser facilmente classificados. A importância do balanço entre taxa de erro e taxa de rejeição foi destacada, enfatizando que a opção de rejeição pode reduzir o erro geral ao custo de aumentar a rejeição de padrões.

A seção 3.2 detalhou a regra de decisão ótima proposta por Chow, que visa minimizar a taxa de erro em relação a uma taxa de rejeição específica. A regra foi formalizada matematicamente e discutida em termos de suas implicações práticas e limitações, como o impacto dos custos de rejeição, erro e acerto no desempenho do sistema.

As abordagens para implementar a classificação com opção de rejeição foram exploradas, incluindo o uso de um único classificador binário, dois classificadores independentes, e a integração da opção de rejeição diretamente no classificador.

O capítulo conclui com uma discussão sobre a extensão da técnica para problemas de classificação multi-classe e apresentação dos modelos utilizado neste trabalho, onde o conceito de rejeição é aplicado considerando múltiplas classes e suas respectivas medidas de confiança e avaliação através da curva ARC.

4 Comitês de classificadores com opção de rejeição

Este capítulo investiga o uso de comitês de classificadores combinados com a estratégia de opção de rejeição, empregando o índice de Gini para aprimorar a capacidade de classificação e ajuste de custo em problemas de classificação de padrões. A estratégia de opção de rejeição permite ao sistema optar por não atribuir uma classe quando a confiança na classificação é baixa, melhorando assim a tomada de decisão em situações incertas. Os comitês utilizam múltiplos modelos, onde cada membro contribui com suas decisões individuais e a abordagem de rejeição é aplicada para lidar com incertezas.

Duas propostas principais são apresentadas: a primeira aplica a opção de rejeição diretamente à saída do classificador, enquanto a segunda aplica a opção de rejeição nas saídas do comitê. Ambas as abordagens são detalhadas com exemplos de implementação e algoritmos, proporcionando uma visão clara sobre como a rejeição pode ser integrada tanto nos classificadores individuais quanto na saída do comitê de classificadores para efeito de comparação entre as abordagens.

4.1 Classificação com Opção de Rejeição (CRO) - Proposta 1

Nesta proposta, apresentamos uma abordagem para implementar a tomada de decisão de qualquer classificador com saída probabilística por meio da aplicação do índice de Gini diretamente à saída do modelo. Essa abordagem visa identificar casos de baixa confiança e fornecer uma opção de rejeição para decisões mais precisas, considerando o custo de rejeição na tarefa de classificação.

Figura 5 – Classificação com Opção de Rejeição.



Fonte: Autor

4.1.1 Geração de Saídas dos modelos

Seja $\mathcal{D}$ o conjunto de dados com n amostras. Um classificador que produz probabilidades para cada classe pode ser descrito da seguinte forma:

Para uma amostra $\mathbf{x}_i$ do conjunto de teste, a probabilidade predita de que $\mathbf{x}_i$ pertença a cada uma das C classes é dada por um vetor de probabilidades $P(C_j|\mathbf{x}_i)$:

$$P(C_j|\mathbf{x}_i) = (p_{i1}, p_{i2}, \dots, p_{iC}),$$

onde p_{ij} é a probabilidade predita da i -ésima amostra pertencer à j -ésima classe.

4.1.2 Tomada de Decisão com Opção de Rejeição

Aplicamos o índice de Gini diretamente ao vetor de probabilidades $P(C_j|\mathbf{x}_i)$ para avaliar a confiança ou incerteza da classificação. O índice de Gini para cada amostra x_i é calculado por:

$$IG = 1 - \sum_{j=1}^k p_{ij}^2, \quad (4.1)$$

onde k é o número de classes.

Definimos o limiar t para o índice de Gini através da equação:

$$t = \frac{k-1}{k}. \quad (4.2)$$

Se $IG > t$, a classificação é considerada incerta e a amostra $\mathbf{x}_i$ é rejeitada. Caso contrário, seleciona-se a classe j com a maior probabilidade em $P(C_j|\mathbf{x}_i)$. O limiar t é baseado na Regra de Decisão Ótima de CHOW e deve ser proporcional ao número de classes C .

Esses valores refletem a ideia de que quanto mais próximo o índice de Gini estiver de t , maior a incerteza na classificação. Para casos binários, o limiar baseado no índice de Gini varia de 0.0 a 0.5, onde 0.5 indica máxima incerteza e 0 indica certeza total. Assim, um limiar pode ser utilizado para determinar se a rejeição deve ser aplicada:

$$\hat{y}(\mathbf{x}) = \begin{cases} \text{Rejeitar,} & \text{se } IG > t \\ \text{Classe } j, & \text{caso contrário} \end{cases}$$

Portanto, para cada instância do conjunto de teste, o algoritmo avalia o índice de Gini em comparação com o limiar definido para decidir se a instância deve ser rejeitada ou classificada definindo assim o Algoritmo 3.

Algoritmo 3 Classificador com Opção de Rejeição (CRO) - Proposta 1

-
- 1: **Procedimento** CRO($P(C_j|\mathbf{x}_i), t$)
 - 2: Calcular o índice de Gini IG para $P(C_j|\mathbf{x}_i)$ (Equação 4.1)
 - 3: **Se** $IG > t$ **Então**
 - 4: Rejeitar (-1)
 - 5: **Senão**
 - 6: Classificar $P(C_j|\mathbf{x}_i)$ ▷ classe de maior probabilidade
 - 7: **Fim Se**
-

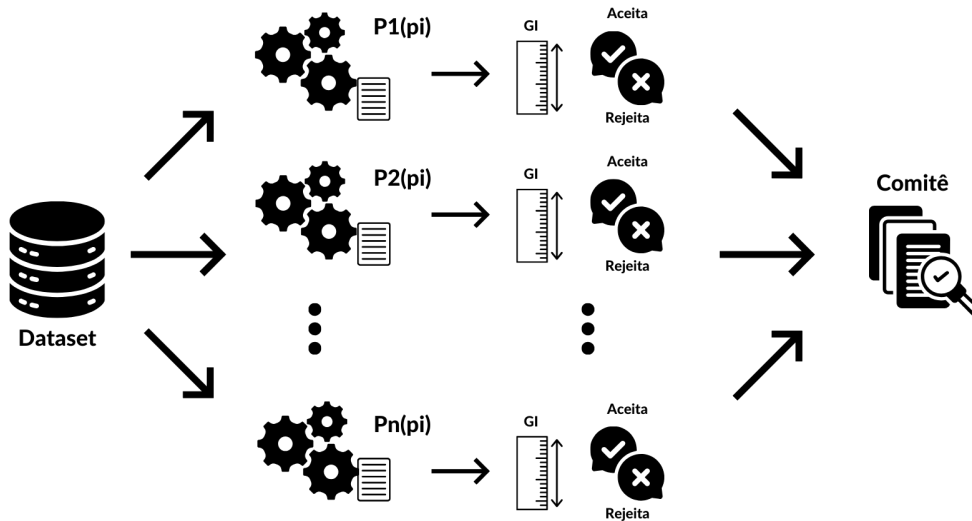
4.2 Comitê de classificadores com opção de rejeição

4.2.1 Votação com Opção de Rejeição (VRO) - Proposta 2

Nesta proposta, exploramos a utilização de comitês como classificadores, onde cada membro do ensemble gera probabilidades associadas às diferentes classes de um padrão de entrada fornecido. Essas probabilidades refletem a confiança de cada membro do ensemble em relação à sua respectiva classificação.

A fim de quantificar essa confiança e habilitar a tomada de decisões informadas, incorporamos a métrica de impureza de Gini ao processo (subseção 4.3.1). Para a nova proposta, vamos decidir a classificação usando a tomada de decisão do Classificador de voto majoritário (Figura 6).

Figura 6 – Votação com Opção de Rejeição (VRO).



Fonte: Autor

4.2.2 Geração das Saídas do Comitê

Para cada entrada $\mathbf{x}$ no conjunto de teste, ativamos n modelos distintos $\mathbf{m}_1, \mathbf{m}_2, \dots, \mathbf{m}_n$ pertencentes ao comitê. Cada modelo $\mathbf{m}_j$ gera uma saída $P(C_i|\mathbf{x})$, que representa a estimativa de probabilidade de que a entrada $\mathbf{x}$ pertença à classe C_i de acordo com o modelo

$\mathbf{m}_j$. Em outras palavras, $P(C_i|\mathbf{x})$ é a probabilidade estimada pelo j -ésimo modelo $\mathbf{m}_j$ de que o padrão $\mathbf{x}$ pertença à classe C_i . Essas saídas são combinadas no vetor de saída do comitê $\mathbf{D}$:

$$\mathbf{D} = [P(C_1|\mathbf{x})_{\mathbf{m}_1}, P(C_2|\mathbf{x})_{\mathbf{m}_2}, \dots, P(C_k|\mathbf{x})_{\mathbf{m}_n}],$$

onde $P(C_i|\mathbf{x})_{\mathbf{m}_j}$ é a probabilidade estimada pelo j -ésimo modelo $\mathbf{m}_j$ de que $\mathbf{x}$ pertença à classe C_i .

A principal distinção em relação à primeira proposta reside na introdução da opção de rejeição na tomada de decisão do Voto majoritário. Essa opção avalia se um padrão deve ser rejeitado ou classificado, com base no limiar de rejeição definido pelo índice de Gini (IG), como expresso na Equação 4.1:

$$\hat{y}(\mathbf{x}) = \begin{cases} \text{Rejeitar,} & \text{se } IG > t \\ \text{Classe } i, & \text{caso contrário} \end{cases}$$

Consequentemente, a saída final do modelo ensemble consistirá em uma saída discreta com o resultado da classificação.

Algoritmo 4 Votação com Opção de Rejeição (VRO)

- Proposta 2

- 1: **Procedimento** VRO($\mathbf{D}, t$)
 - 2: **Inicializar** um vetor vazio resultados
 - 3: **Para** cada vetor de probabilidade $P(C_i|\mathbf{x})$ em $\mathbf{D}$ **faça**
 - 4: $resultado \leftarrow \text{CRO}(P(C_i|\mathbf{x}), t)$ ▷ CRO - classificador individual
 - 5: Adicionar *resultado* ao vetor resultados
 - 6: **Fim Para**
 - 7: **Calcular** a moda dos valores em resultados
 - 8: **Retornar** a classe com maior frequência ou -1 se houver empate
-

Outra forma de demonstrar a saída através é da seguinte tabela:

$\mathbf{x}_i$	Modelo 1		Modelo 2		Modelo 3		Decisão
	$P(C_i \mathbf{x})$	$\hat{y}$	$P(C_i \mathbf{x})$	$\hat{y}$	$P(C_i \mathbf{x})$	$\hat{y}$	
$\mathbf{x}_1$	[0.8, 0.2]	0	[0.4, 0.6]	-1	[0.6, 0.4]	-1	-1
$\mathbf{x}_2$	[0.3, 0.7]	1	[0.6, 0.4]	-1	[0.5, 0.5]	-1	-1
$\mathbf{x}_3$	[0.6, 0.4]	-1	[0.2, 0.8]	1	[0.3, 0.7]	1	1
$\mathbf{x}_4$	[0.9, 0.1]	0	[0.7, 0.3]	0	[0.8, 0.2]	0	0

Tabela 1 – Exemplo de Classificação com VRO

4.3 Conclusão

Este capítulo investiga o uso de comitês de classificadores combinados com a estratégia de opção de rejeição para aprimorar a capacidade de classificação e ajuste de custo em problemas de classificação de padrões. A estratégia de opção de rejeição permite ao sistema optar por não atribuir uma classe quando a confiança na classificação é baixa, melhorando a tomada de decisão em situações incertas.

São apresentadas duas propostas principais:

4.3.1 Classificação com Opção de Rejeição (CRO) - Proposta 1

A primeira proposta aplica a opção de rejeição diretamente à saída do classificador. Utiliza o índice de Gini para avaliar a confiança na classificação. Se o índice de Gini exceder um limiar definido, a classificação é rejeitada. Caso contrário, a classe com a maior probabilidade é escolhida. O algoritmo é detalhado e ilustrado com um exemplo prático.

4.3.2 Votação com Opção de Rejeição (VRO) - Proposta 2

A segunda proposta explora o uso de comitês de classificadores, onde cada membro do comitê gera probabilidades para diferentes classes. A opção de rejeição é incorporada na decisão final do comitê, baseada na votação majoritária. O índice de Gini é aplicado para decidir se um padrão deve ser rejeitado ou classificado. A decisão final é determinada pela moda das decisões dos modelos individuais, considerando a possibilidade de rejeição.

5 Resultados e Discussões

Este capítulo descreve a metodologia de avaliação e as simulações computacionais dos métodos propostos no Capítulo 4 aplicados em problemas de multiclassificação. A seção 5.1 descreve a metodologia de experimentos informando os parâmetros de classificação, implementação da opção de rejeição de modo geral, o critério de escolha dos melhores resultados e os conjuntos de dados utilizados. Os resultados dos classificadores são analisados na seção 5.2.

5.1 Metodologia de Experimentos

Nesta seção é apresentada a metodologia de experimentos dos métodos propostos aplicados em problemas de classificação binários e multiclasse. Cada realização, ou seja, o processo de treinamento e teste do modelo foi executado 20 vezes.

5.1.1 Problemas de Classificação

Para avaliar a eficiência das propostas, foram utilizadas diferentes conjuntos de dados obtidas dos repositórios UCI¹, Kaggle² e Sklearn³, a saber: *Breast Cancer Wisconsin*, *Fish Species*, *Iris*, *Handwritten Digits*, *Vertebral Column Patologies* e *Wine*. A Tabela 2 apresenta as principais características desses conjuntos.

Tabela 2 – Resumo dos Conjuntos de Dados

Nome do Dataset	Abrev.	Atributos	Instâncias	Classes
Breast Cancer Wisconsin	BCW	30	569	2
Fish Species	FISH	4	4080	9
Iris	IRIS	4	150	3
Handwritten Digits	MNIST	64	5620	10
Vertebral Column Patologies	VCP	6	310	3
Wine	WINE	13	178	3

Fonte: Autor

Um processo de busca em grade com validação cruzada foi conduzido para ajustar os hiperparâmetros dos classificadores XGB, MLP, SVC, k-NN e RF.

Inicialmente, foram consideradas duas abordagens de padronização dos dados: *StandardScaler* e *MinMaxScaler*. A escolha entre essas abordagens foi realizada para

¹ Disponível em: <<https://archive.ics.uci.edu/>>

² Disponível em: <<https://www.kaggle.com/datasets>>

³ Disponível em: <https://scikit-learn.org/stable/datasets/toy_dataset.html>

avaliar seu impacto no desempenho dos modelos. O *StandardScaler* normaliza os dados para que tenham média zero e desvio padrão um, enquanto o *MinMaxScaler* ajusta os dados para que fiquem dentro de um intervalo específico, geralmente entre 0 e 1.

Para cada conjunto de dados, os dados foram inicialmente divididos em dois subconjuntos: um conjunto de treinamento e um conjunto de teste, com uma divisão de 80% para treinamento e 20% para teste.

A busca em grade foi aplicada a cada classificador, com a definição de um *grid* de parâmetros específicos para cada modelo. A validação cruzada com 5 folds foi utilizada para avaliar o desempenho de cada combinação de parâmetros.

Durante o processo de busca em grade, o desempenho de cada combinação de parâmetros foi avaliado utilizando métricas de acurácia. A configuração que proporcionou o melhor desempenho no conjunto de dados específico foi selecionada como a ideal para cada classificador.

É importante notar que o classificador *Gaussian Naive Bayes* (GNB) não possui hiperparâmetros ajustáveis. Portanto, não foi necessário realizar busca em grade para este modelo, pois ele não requer ajuste de parâmetros além dos próprios dados de entrada.

A seguir, são apresentados os parâmetros otimizados para cada modelo e conjunto de dados.

Tabela 3 – Hiper-parâmetros para o XGB

Parâmetro	BCW	FISH	IRIS	MNIST	VCP	WINE
colsample_bytree	0.8	1.0	1.0	0.8	1.0	0.8
learning_rate	0.2	0.05	0.01	0.1	0.1	0.01
max_depth	5	5	3	3	3	3
n_estimators	200	100	200	200	100	50
subsample	0.8	0.8	1.0	0.8	0.8	0.8

Fonte: Autor

Tabela 4 – Hiper-parâmetros para o MLP

Parâmetro	BCW	FISH	IRIS	MNIST	VCP	WINE
activation	tanh	relu	tanh	relu	tanh	tanh
alpha	0.001	1e-05	0.001	0.001	0.001	0.001
hidden_layer_sizes	(64,)	(64,)	(64,)	(64,)	(100,)	(64,)
learning_rate	constant	constant	constant	constant	constant	constant
max_iter	2000	2000	2000	2000	2000	2000

Fonte: Autor

Tabela 5 – Hiper-parâmetros para o SVC

Parâmetro	BCW	FISH	IRIS	MNIST	VCP	WINE
C	0.1	100	10	10	10	1
gamma	scale	scale	scale	scale	scale	scale
kernel	linear	rbf	linear	poly	linear	sigmoid

Fonte: Autor

Tabela 6 – Hiper-parâmetros para o K-NN

Parâmetro	BCW	FISH	IRIS	MNIST	VCP	WINE
n_neighbors	5	7	11	3	11	5
metric	manhattan	euclidean	euclidean	manhattan	manhattan	manhattan
weights	uniform	uniform	distance	distance	distance	uniform

Fonte: Autor

Tabela 7 – Hiper-parâmetros para o RF

Parâmetro	BCW	FISH	IRIS	MNIST	VCP	WINE
criterion	entropy	entropy	gini	gini	gini	gini
max_depth	10	7	5	10	10	3
min_samples_leaf	1	2	2	1	1	1
min_samples_split	2	5	2	5	10	10
n_estimators	100	200	150	200	150	100

Fonte: Autor

5.1.2 Parâmetros de Classificação com Opção de Rejeição

Como dito, a classificação com opção de rejeição em comitê utiliza como base o índice de Gini (GI). O algoritmo é inicializado com três parâmetros principais: número de classes distintas no problema de classificação, número de variáveis (ou divisões) para inicialmente determinar os valores de rejeição baseados no número de classes e o conjunto de predições probabilísticas fornecidas pelos componentes do comitê.

A lista de valores dos pesos de t , é definida com base no valor máximo possível do GI (Equação (4.2)) e o dividimos pelo número de variáveis. Neste trabalho, optamos por utilizar 12 elementos. Os limiares de rejeição são utilizados para determinar quando uma amostra deve ser rejeitada. Os limiares de rejeição para cada custo de rejeição w_r variam de acordo com a quantidade de classes: para um problema binário, a variação é de 0 a 0,5; para um problema com três classes, a variação é de 0 a 0,67.

Com isso, podemos calcular os valores do GI para cada previsão probabilística fornecida pelos classificadores do ensemble utilizando a Equação (4.1), através das probabilidades associadas a cada classe.

Para realizar a classificação com opção de rejeição, aplicamos a rejeição a uma predição com base em um limiar de rejeição w_r específico. Se o índice de Gini da predição

for maior que o valor de rejeição, a predição é rejeitada (-1); caso contrário, a classe com a maior probabilidade é considerada como rótulo. A determinação da classe é feita por votação majoritária; no entanto, se houver empate na votação, a amostra é rejeitada.

5.1.3 Avaliação dos Modelos

Para a avaliação dos modelos, foram utilizadas duas métricas principais: a média de acerto e a média de rejeição obtida no subconjunto de teste. A média de acerto é calculada como a média das taxas de acerto nos testes ao longo das execuções. A taxa de acerto de cada execução é definida como a proporção de instâncias classificadas corretamente em relação ao total de instâncias no subconjunto de teste, excluindo os padrões rejeitados.

A média de rejeição corresponde à média das taxas de rejeição observadas nas execuções. A taxa de rejeição de cada execução é calculada como a proporção de instâncias rejeitadas em relação ao total de instâncias no subconjunto de teste.

A curva AR representa o desempenho dos classificadores, mantendo uma forma curva que reflete os resultados obtidos. Cada ponto da curva AR corresponde à média de acerto e rejeição para um determinado custo de rejeição w_r . Para mais detalhes, veja a Equação (3.15).

5.2 Resultados

Nesta seção, são apresentados os resultados dos experimentos realizados com quatro conjuntos de dados distintos: BCW, FISH, IRIS, MNIST, VCP e WINE. Em cada conjunto de dados, foram avaliados sete classificadores, identificados como RO_MLP, RO_KNN, RO_XGB, RO_RF, RO_SVM, RO_GNB e RO_VOTING. Para efeito de comparação, foram definidos os modelos individuais RO_MLP, RO_KNN, RO_XGB, RO_SVM e RO_GNB, além do modelo em comitê RO_RF como parte da Proposta 1. Para a Proposta 2, foi considerado o comitê RO_VOTING, composto por todos os classificadores da Proposta 1, ao qual foi aplicada a abordagem de classificação com opção de rejeição para comitês.

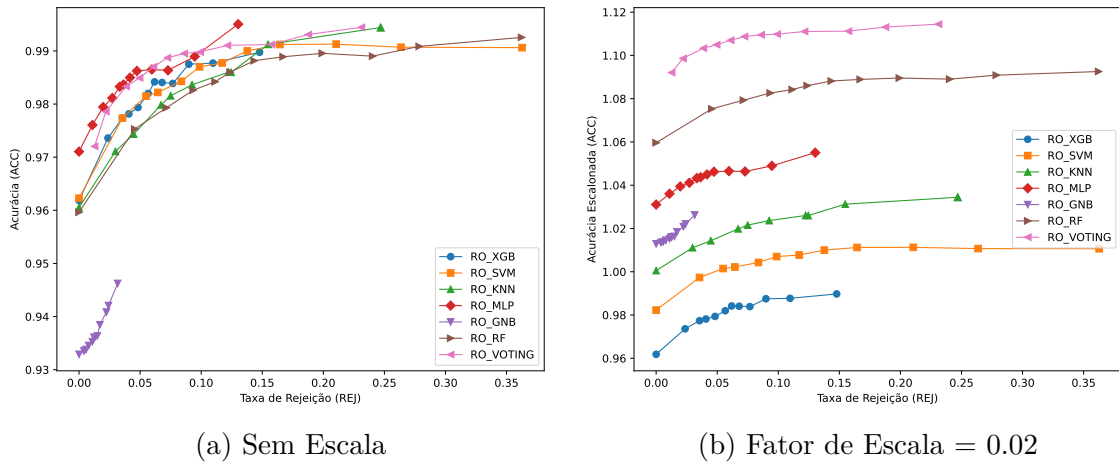
Os resultados foram organizados em subseções que analisam o desempenho dos modelos em termos da curva de Acurácia e Rejeição (AR), do Risco Empírico e, por fim, apresentam uma tabela geral que sumariza os resultados com base nos diferentes valores de w_r . A análise comparativa entre o comitê RO_VOTING e os modelos individuais permite avaliar a eficácia da estratégia de rejeição na melhoria do desempenho dos classificadores em diferentes cenários.

5.2.1 Conjunto de dados BCW

5.2.1.1 Desempenho na Curva AR em BCW

A curva de Acurácia e Rejeição (AR) para o conjunto de dados BCW é apresentada na Figura (7a) apresenta a variação da acurácia média e da taxa de rejeição média dos classificadores, enquanto a Figura (7b) mostra essas mesmas variações, mas com a aplicação de um fator de escala na reta da acurácia para melhor visualização dos resultados.

Figura 7 – Relação entre Taxas de Acurácia e Reieição - BCW



Fonte: Autor.

A curva AR de todos os modelos segue o padrão esperado para o experimento, com exceção do comportamento observado no RO_VOTING. Este desvio ocorre porque o RO_VOTING implementa um passo adicional no processo de rejeição, ativado quando há empates na decisão. Nessas situações, ao invés de escolher uma das opções empatadas, o classificador opta por rejeitar a amostra.

Em termos de desempenho geral, o RO_VOTING ainda se destaca como o classificador mais eficaz, seguido de perto pelo RO_RF e RO_MLP, como mostrado na Figura (7b). No entanto, o modelo RO_XGB apresentou o pior desempenho, indicando que ele tem mais dificuldade em manter um equilíbrio eficiente entre acurácia e rejeição.

Para proporcionar uma visão global do desempenho dos classificadores no conjunto de dados BCW, a Tabela 8 apresenta os resultados com base em diferentes valores de w_r em todos os modelos.

Tabela 8 – Acurácia e Rejeição (ACC% e REJ%) em BCW

w_r	RO_XGB		RO_SVM		RO_KNN		RO_MLP		RO_GNB		RO_RF		RO_VOTING	
	ACC%	REJ%	ACC%	REJ%	ACC%	REJ%	ACC%	REJ%	ACC%	REJ%	ACC%	REJ%	ACC%	REJ%
0.04	98.98	14.78	99.06	36.27	99.44	24.69	99.50	13.03	94.62	3.16	99.25	36.23	99.44	23.11
0.08	98.77	10.96	99.07	26.36	99.44	24.69	98.90	9.47	94.21	2.41	99.09	27.85	99.31	18.77
0.12	98.75	8.99	99.13	21.05	99.44	24.69	98.64	7.28	94.08	2.24	98.90	24.04	99.12	15.75
0.17	98.39	7.68	99.12	16.45	99.12	15.48	98.65	5.96	93.84	1.71	98.95	19.91	99.11	12.15
0.21	98.41	6.80	99.00	13.77	98.60	12.46	98.62	4.74	93.63	1.49	98.89	16.71	98.98	9.91
0.25	98.42	6.18	98.77	11.71	98.60	12.24	98.50	4.17	93.60	1.27	98.82	14.34	98.95	8.60
0.29	98.20	5.66	98.70	9.87	98.36	9.25	98.37	3.64	93.61	1.23	98.60	12.37	98.87	7.24
0.33	97.94	4.82	98.43	8.38	98.16	7.50	98.33	3.33	93.53	1.10	98.42	11.14	98.70	6.10
0.38	97.81	4.08	98.22	6.45	97.98	6.71	98.11	2.72	93.46	0.79	98.26	9.34	98.49	4.96
0.42	97.74	3.55	98.15	5.48	97.43	4.47	97.94	1.97	93.38	0.57	97.93	7.15	98.33	3.86
0.46	97.36	2.37	97.73	3.55	97.11	2.98	97.61	1.10	93.35	0.39	97.53	4.56	97.86	2.19
0.50	96.18	0.00	96.23	0.00	96.05	0.00	97.11	0.00	93.29	0.00	95.96	0.00	97.20	1.27

Fonte: Autor

5.2.1.2 Análise dos Classificadores em Relação ao Risco Empírico em BCW

Para uma avaliação mais crítica dos classificadores, consideramos o valor do Risco Empírico (REMP) associado ao classificador de votação (RO_VOTING) para três diferentes configurações de w_r : 0.08, 0.29 e 0.50.

Tabela 9 – Rendimento dos classificadores para $w_r = 0.08$

Classificador	ACC (%)	ERR (%)	REJ (%)	REMP
RO_MLP	98.90	1.10	9.47	0.0186
RO_XGB	98.77	1.23	10.96	0.0211
RO_VOTING	99.31	0.69	18.77	0.0219
RO_KNN	99.44	0.56	24.69	0.0253
RO_SVM	99.07	0.93	26.36	0.0304
RO_RF	99.09	0.91	27.85	0.0314
RO_GNB	94.21	5.79	2.41	0.0599

Fonte: Autor

Na Tabela 9, os resultados indicam que o classificador RO_VOTING possui um REMP de 0.0219. Embora o modelo RO_MLP, com um REMP de 0.0186, apresente um desempenho ligeiramente superior em termos de minimização do risco, o RO_VOTING ainda supera outros classificadores como RO_SVM e RO_RF, que têm REMP de 0.0304 e 0.0314, respectivamente. Por outro lado, o modelo RO_GNB, com um REMP de 0.0599, demonstra um desempenho menos eficiente, refletindo sua alta taxa de rejeição.

Tabela 10 – Rendimento dos classificadores para $w_r = 0.29$

Classificador	ACC (%)	ERR (%)	REJ (%)	REMP
RO_MLP	98.37	1.63	3.64	0.0269
RO_VOTING	98.87	1.13	7.24	0.0323
RO_SVM	98.70	1.30	9.87	0.0416
RO_KNN	98.36	1.64	9.25	0.0432
RO_XGB	98.20	1.80	5.66	0.0344
RO_RF	98.60	1.40	12.37	0.0499
RO_GNB	93.61	6.39	1.23	0.0675

Fonte: Autor

Para a configuração com $w_r = 0.29$, mostrada na Tabela 10, o RO_VOTING tem um REMF de 0.0323, que é inferior ao de RO_RF (0.0499) e RO_GNB (0.0675), mas superior ao de RO_MLP, que apresenta um REMF de 0.0269. O RO_VOTING se destaca como uma melhor opção, principalmente quando comparado a RO_KNN e RO_SVM, que têm REMF de 0.0432 e 0.0416, respectivamente.

Tabela 11 – Rendimento dos classificadores para $w_r = 0.50$

Classificador	ACC (%)	ERR (%)	REJ (%)	REMP
RO_MLP	97.11	2.89	0.00	0.0289
RO_VOTING	97.20	2.80	1.27	0.0343
RO_SVM	96.23	3.77	0.00	0.0377
RO_XGB	96.18	3.82	0.00	0.0382
RO_KNN	96.05	3.95	0.00	0.0395
RO_RF	95.96	4.04	0.00	0.0404
RO_GNB	93.29	6.71	0.00	0.0671

Fonte: Autor

Na Tabela 11, o RO_VOTING apresenta um REMF de 0.0343. Embora esse valor seja ligeiramente maior ao do RO_MLP (0.0289), se mostra melhor do que o RO_XGB e o RO_SVM, com REMF de 0.0382 e 0.0377, respectivamente. O RO_VOTING também apresenta uma taxa de rejeição de 1.27%, mostrando que o método de critério de desempate foi acionado. A capacidade do RO_VOTING de equilibrar a precisão da classificação com uma taxa de rejeição eficiente demonstra seu desempenho competitivo, mesmo apresentando rejeição de padrões em comparação com outros modelos que não apresentaram.

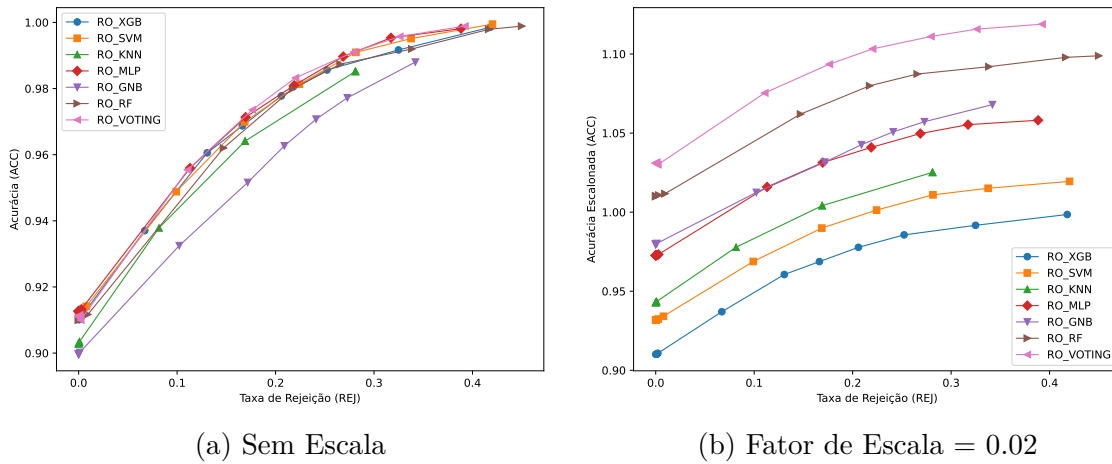
Essa análise detalha o desempenho de cada classificador com base no Risco Empírico para o conjunto de dados BCW. O RO_VOTING se destaca por sua performance robusta, especialmente quando comparado a classificadores que não conseguiram equilibrar precisão e rejeição de forma tão eficaz.

5.2.2 Conjunto de dados FISH

5.2.2.1 Desempenho na Curva AR em FISH

A curva de Acurácia e Rejeição (AR) para o conjunto de dados FISH é apresentada na Figura (8a) apresenta a variação da acurácia média e da taxa de rejeição média dos classificadores, enquanto a Figura (8b) mostra essas mesmas variações, mas com a aplicação de um fator de escala na reta da acurácia para melhor visualização dos resultados.

Figura 8 – Relação entre Taxas de Acurácia e Rejeição - FISH



Fonte: Autor.

Em termos de desempenho geral, o RO_VOTING se apresenta como o classificador mais eficaz, seguido de perto pelo RO_RF e RO_MLP que um certo ponto é ultrapassado pelo RO_GNB, como mostrado na Figura (8b). No entanto, o modelo RO_XGB apresentou o pior desempenho, indicando que ele tem mais dificuldade em manter um equilíbrio eficiente entre acurácia e rejeição.

Para proporcionar uma visão global do desempenho dos classificadores no conjunto de dados FISH, a Tabela 12 apresenta os resultados com base em diferentes valores de w_r em todos os modelos.

Tabela 12 – Acurácia e Rejeição (ACC% e REJ%) em FISH

w_r	RO_XGB		RO_SVM		RO_KNN		RO_MLP		RO_GNB		RO_RF		RO_VOTING	
	ACC%	REJ%	ACC%	REJ%	ACC%	REJ%	ACC%	REJ%	ACC%	REJ%	ACC%	REJ%	ACC%	REJ%
0.07	99.85	41.80	99.95	41.94	98.52	28.11	99.79	38.73	98.80	34.20	99.88	45.04	99.87	39.17
0.15	99.17	32.48	99.48	33.64	98.52	28.11	99.49	31.53	97.72	27.29	99.80	41.84	99.51	32.49
0.22	98.56	25.23	99.08	28.10	98.52	28.11	98.92	26.74	97.07	24.12	99.24	34.23	99.07	27.87
0.30	97.78	20.59	98.13	22.32	96.41	16.91	98.09	21.78	96.27	20.89	98.86	26.73	98.38	22.15
0.37	96.87	16.62	97.04	16.88	96.41	16.91	96.98	16.80	95.16	17.17	98.00	21.62	97.31	17.59
0.44	96.06	13.06	94.86	9.90	93.79	8.16	95.57	11.13	93.25	10.22	96.27	14.55	95.56	11.05
0.52	93.70	6.72	91.44	0.81	90.35	0.09	91.40	0.27	90.02	0.11	91.07	0.95	90.98	0.20
0.59	91.08	0.22	91.24	0.25	90.33	0.07	91.36	0.08	89.96	0.00	90.98	0.33	91.17	0.10
0.67	91.03	0.12	91.20	0.07	90.29	0.00	91.31	0.00	89.96	0.00	90.91	0.07	91.13	0.01
0.74	91.02	0.09	91.18	0.03	90.29	0.00	91.31	0.00	89.96	0.00	90.89	0.00	91.10	0.00
0.81	91.02	0.09	91.18	0.00	90.29	0.00	91.31	0.00	89.96	0.00	90.89	0.00	91.10	0.00
0.89	91.01	0.00	91.18	0.00	90.29	0.00	91.31	0.00	89.96	0.00	90.89	0.00	91.10	0.00

Fonte: Autor

5.2.2.2 Análise dos Classificadores em Relação ao Risco Empírico em FISH

Para uma avaliação mais crítica dos classificadores, consideramos o valor do Risco Empírico (REMP) associado ao classificador de votação (RO_VOTING) para três diferentes configurações de w_r : 0.15, 0.44 e 0.81.

Tabela 13 – Rendimento dos classificadores para $w_r = 0.15$

Classificador	ACC (%)	REJ (%)	ERR (%)	REMP
RO_MLP	0.99	0.01	0.32	0.0524
RO_VOTING	0.99	0.00	0.32	0.0536
RO_SVM	0.99	0.01	0.34	0.0556
RO_XGB	0.99	0.01	0.32	0.0570
RO_KNN	0.99	0.01	0.28	0.0570
RO_GNB	0.98	0.02	0.27	0.0637
RO_RF	1.00	0.00	0.42	0.0648

Fonte: Autor

Os dados apresentados na Tabela 13 revelam que o classificador RO_VOTING apresenta um REMP de 0.0536. Embora o modelo RO_MLP tenha um REMP ligeiramente inferior, de 0.0524, e demonstre um desempenho marginalmente melhor na minimização do risco, o RO_VOTING se destaca em comparação com outros classificadores, como RO_SVM e RO_XGB, que apresentam REMP de 0.0556 e 0.0570, respectivamente. Em contrapartida, o modelo RO_GNB, com um REMP de 0.0648, indica um desempenho menos eficiente, refletindo uma taxa de rejeição mais elevada. Essa análise sugere que, embora todos os modelos apresentem desempenhos competitivos, o RO_MLP se destaca na redução do risco empírico, seguido de perto pelo RO_VOTING.

Para a configuração com $w_r = 0.44$, apresentada na Tabela 14, o classificador RO_VOTING apresenta um REMP de 0.0931, superior a todos os outros, seguido pelo RO_MLP, que possui um REMP de 0.0933. Ambos os modelos têm desempenhos muito próximos em termos de risco empírico. O RO_SVM, por sua vez, apresenta um REMP de

Tabela 14 – Rendimento dos classificadores para $w_r = 0.44$

Classificador	ACC (%)	ERR (%)	REJ (%)	REMP
RO_VOTING	0.96	0.04	0.11	0.0931
RO_MLP	0.96	0.04	0.11	0.0933
RO_SVM	0.95	0.05	0.10	0.0949
RO_XGB	0.96	0.04	0.13	0.0969
RO_KNN	0.94	0.06	0.08	0.0981
RO_RF	0.96	0.04	0.15	0.1014
RO_GNB	0.93	0.07	0.10	0.1125

Fonte: Autor

0.0949, tornando-se uma opção viável, embora um pouco menos eficiente em comparação com os modelos anteriores.

Os classificadores RO_XGB e RO_RF apresentam REMF de 0.0969 e 0.1014, respectivamente, o que indica que eles têm um desempenho inferior em relação ao RO_VOTING e RO_MLP, evidenciando uma maior ineficiência na minimização do risco empírico. Por outro lado, o RO_GNB apresenta REMF de 0.1125 revelando um desempenho ainda menos eficaz.

Tabela 15 – Rendimento dos classificadores para $w_r = 0.81$

Classificador	ACC (%)	ERR (%)	REJ (%)	REMP
RO_MLP	0.91	0.09	0.00	0.0869
RO_SVM	0.91	0.09	0.00	0.0882
RO_VOTING	0.91	0.09	0.00	0.0890
RO_XGB	0.91	0.09	0.00	0.0905
RO_RF	0.91	0.09	0.00	0.0911
RO_KNN	0.90	0.10	0.00	0.0971
RO_GNB	0.90	0.10	0.00	0.1004

Fonte: Autor

Na Tabela 15, o classificador RO_MLP apresenta um REMF de 0.0869, o que o coloca como o melhor entre os avaliados. O RO_SVM, com um REMF de 0.0882, é o segundo melhor classificador, seguido pelo RO_VOTING, que tem um REMF de 0.0890. Esses três modelos demonstram uma capacidade superior de minimizar o risco empírico em comparação aos demais.

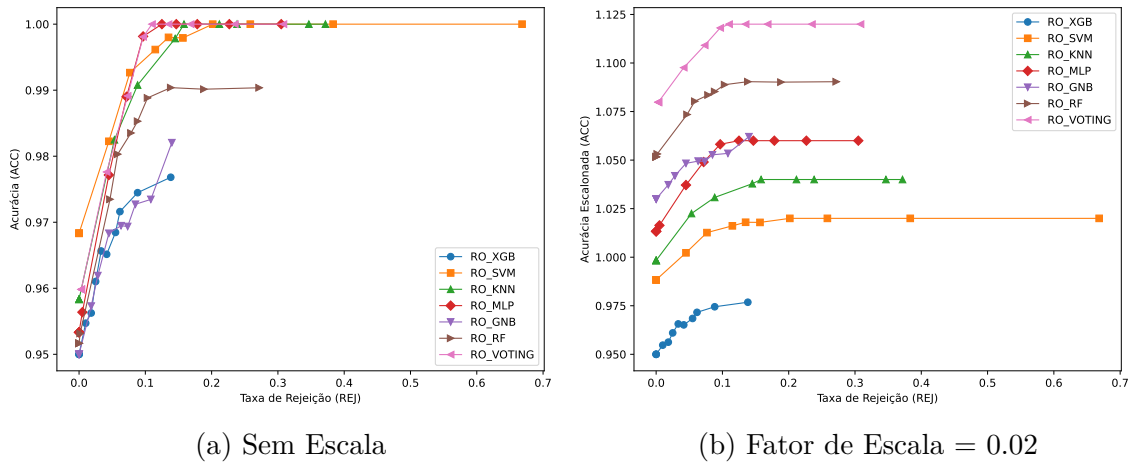
Em contrapartida, os três piores classificadores são o RO_GNB, com um REMF de 0.1004, o RO_KNN, que apresenta um REMF de 0.0971, e o RO_RF, que tem um REMF de 0.0911. Esses resultados indicam que esses modelos têm desempenho inferior na minimização do risco, refletindo uma maior ineficiência em suas classificações.

5.2.3 Conjunto de dados IRIS

5.2.3.1 Desempenho na Curva AR em IRIS

A curva de Acurácia e Rejeição (AR) para o conjunto de dados IRIS é apresentada nas A Figura 9a apresenta a variação da acurácia média e da taxa de rejeição média dos classificadores, enquanto a Figura 9b mostra essas mesmas variações, mas com a aplicação de um fator de escala na reta da acurácia para melhor visualização dos resultados.

Figura 9 – Relação entre Taxas de Acurácia e Rejeição - IRIS



Fonte: Autor.

A curva AR de todos os modelos segue o padrão esperado para o experimento assim como o experimento anterior, com exceção do comportamento observado no RO_VOTING já explicado nos resultados anteriores.

Em termos de desempenho geral, o RO_VOTING ainda se destaca como o classificador mais eficaz, seguido de perto pelo RO_RF, RO_MLP e RO_GNB, como mostrado na Figura 9b. No entanto, o modelo RO_XGB apresentou o pior desempenho na maior parte observável.

Para proporcionar uma visão global do desempenho dos classificadores no conjunto de dados IRIS, a Tabela 16 apresenta os resultados com base em diferentes valores de w_r em todos os modelos.

Tabela 16 – Acurácia e Rejeição (ACC% e REJ%) em IRIS

w_r	RO_XGB		RO_SVM		RO_KNN		RO_MLP		RO_GNB		RO_RF		RO_VOTING	
	ACC%	REJ%	ACC%	REJ%	ACC%	REJ%	ACC%	REJ%	ACC%	REJ%	ACC%	REJ%	ACC%	REJ%
0.06	97.68	13.83	100.0	66.83	100.0	37.17	100.0	30.50	98.20	14.00	99.04	27.17	100.0	30.83
0.11	97.45	8.83	100.0	38.33	100.0	34.67	100.0	22.67	97.35	10.83	99.01	18.83	100.0	23.50
0.17	97.16	6.17	100.0	25.83	100.0	23.83	100.0	17.83	97.27	8.50	99.04	13.83	100.0	16.83
0.22	96.85	5.50	100.0	20.17	100.0	21.17	100.0	14.67	96.94	7.33	98.88	10.33	100.0	13.50
0.28	96.51	4.17	99.79	15.67	100.0	15.83	100.0	12.50	96.95	6.33	98.53	8.83	100.0	11.00
0.33	96.56	3.33	99.80	13.50	99.78	14.50	99.81	9.67	96.83	4.50	98.35	7.83	99.81	9.67
0.39	96.10	2.50	99.62	11.50	99.08	8.83	98.90	7.17	96.19	2.83	98.03	5.83	98.91	7.33
0.44	95.63	1.83	99.27	7.67	98.25	5.33	97.72	4.50	95.73	1.83	97.35	4.67	97.76	4.17
0.50	95.47	1.00	98.23	4.50	95.83	0.00	95.64	0.50	95.00	0.00	95.32	0.17	95.98	0.33
0.56	95.00	0.00	96.83	0.00	95.83	0.00	95.33	0.00	95.00	0.00	95.17	0.00	95.98	0.33
0.61	95.00	0.00	96.83	0.00	95.83	0.00	95.33	0.00	95.00	0.00	95.17	0.00	95.98	0.33
0.67	95.00	0.00	96.83	0.00	95.83	0.00	95.33	0.00	95.00	0.00	95.17	0.00	95.98	0.33

Fonte: Autor

5.2.3.2 Análise dos Classificadores em Relação ao Risco Empírico em IRIS

Para uma avaliação mais crítica dos classificadores, selecionamos as três tabelas mais relevantes com base no valor de Risco Empírico (REMP) associado ao classificador de votação (RO_VOTING). A seguir, apresentamos as análises para as configurações de $w_r = 0.11, 0.39$ e 0.67 .

Tabela 17 – Rendimento dos classificadores para $w_r = 0.11$

Classificador	ACC (%)	REJ (%)	ERR (%)	REMP
RO_MLP	100.0	22.67	0.00	0.0249
RO_VOTING	100.0	23.50	0.00	0.0258
RO_RF	99.01	18.83	0.99	0.0306
RO_XGB	97.45	8.83	2.55	0.0352
RO_KNN	100.0	34.67	0.00	0.0381
RO_GNB	97.35	10.83	2.65	0.0385
RO_SVM	100.0	38.33	0.00	0.0422

Fonte: Autor

A Tabela 17 revela que o classificador RO_VOTING apresenta um REMP de 0.0258. Embora esse valor seja superior ao do RO_MLP, com um REMP de 0.0249, o RO_VOTING ainda demonstra um desempenho com um valor bastante aproximado. Entre os classificadores analisados, o RO_VOTING é um dos três primeiros a apresentar uma taxa de rejeição de 0.00%, o que é uma vantagem comparativa. Mesmo com uma taxa de erro um pouco mais alta, o RO_VOTING mantém um REMP relativamente baixo. Em contraste, o classificador RO_SVM apresenta o maior REMP, de 0.0422, devido a sua taxa de erro mais alta.

A Tabela 18 apresenta os resultados do desempenho dos classificadores para $w_r = 0.39$. O classificador RO_VOTING, com um REMP de 0.0395, exibe um desempenho competitivo quando comparado aos outros modelos. Embora este valor seja ligeiramente

Tabela 18 – Rendimento dos classificadores para $w_r = 0.39$

Classificador	ACC (%)	REJ (%)	ERR (%)	REMP
RO_MLP	98.90	7.17	1.10	0.0389
RO_VOTING	98.91	7.33	1.09	0.0395
RO_KNN	99.08	8.83	0.92	0.0437
RO_RF	98.03	5.83	1.97	0.0424
RO_XGB	96.10	2.50	3.90	0.0487
RO_SVM	99.62	11.50	0.38	0.0487
RO_GNB	96.19	2.83	3.81	0.0491

Fonte: Autor

maior que o REMP do RO_MLP, que obteve 0.0389, e praticamente equivalente ao do RO_RF (0.0424), o RO_VOTING apresenta uma taxa de rejeição de 7.33%.

A análise desses resultados reforça a importância de considerar tanto a acurácia quanto a taxa de rejeição na avaliação da eficácia dos modelos. O RO_VOTING mostra um bom equilíbrio entre esses fatores apresentando valores de REMP muito próximos dos modelos com opção de rejeição na saída do classificador, enquanto o RO_SVM destaca-se por sua alta taxa de rejeição e erro disputando com o RO_GNB na pior posição.

Tabela 19 – Rendimento dos classificadores para $w_r = 0.67$

Classificador	ACC (%)	ERR (%)	REJ (%)	REMP
RO_SVM	96.83	0.00	3.17	0.0317
RO_KNN	95.83	0.00	4.17	0.0417
RO_VOTING	95.98	0.33	4.02	0.0424
RO_MLP	95.33	0.00	4.67	0.0467
RO_RF	95.17	0.00	4.83	0.0483
RO_GNB	95.00	0.00	5.00	0.0500
RO_XGB	95.00	0.00	5.00	0.0500

Fonte: Autor

A Tabela 19 mostra que o classificador RO_VOTING, com um REMP de 0.0424, apresenta um desempenho semelhante aos classificadores sem rejeição, dado que a única taxa de rejeição significativa é observada no RO_VOTING, devido à sua estratégia de rejeição em caso de empates.

Os modelos RO_SVM e RO_KNN apresentam os menores valores de REMP, com 0.0317 e 0.0417, respectivamente, ambos com taxas de erro de 3.17% e 4.17%. Apesar de evitarem a rejeição, apresentam valores de REMP ligeiramente menores em comparação ao RO_VOTING, que possui um REMP de 0.0424.

Por outro lado, os classificadores RO_MLP, RO_RF, RO_GNB e RO_XGB possuem valores de REMP superiores, variando de 0.0467 a 0.0500, com taxas de erro variando de 4.00% a 5.00%. O RO_KNN apresenta o maior REMP entre os modelos

analisados, com 0.0417, e também a maior taxa de erro, com 4.17%.

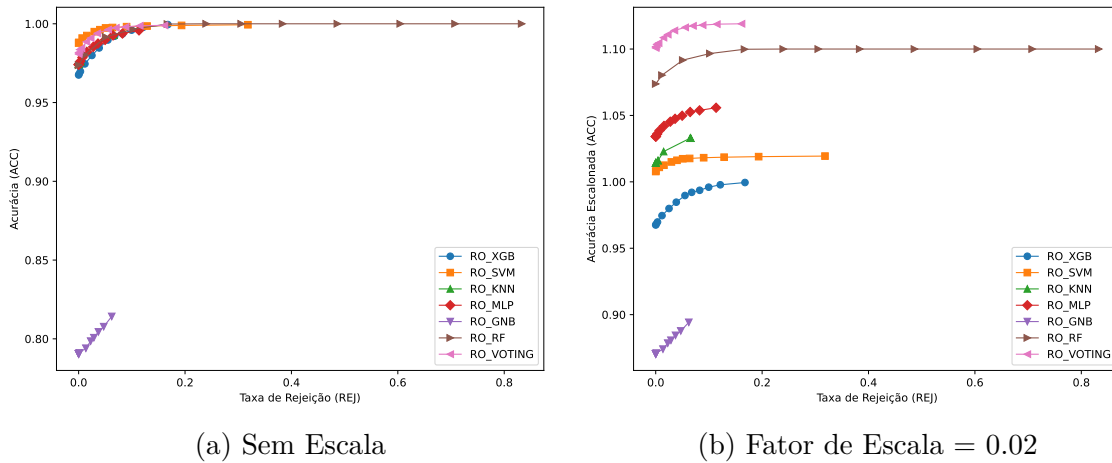
Em conclusão, os classificadores RO_SVM e RO_KNN se destacam em termos de menor risco empírico para $w_r = 0.67$, com o RO_VOTING também mostrando um desempenho competitivo. Já o RO_MLP, RO_RF, RO_GNB e RO_XGB apresentam valores de REMP mais altos, indicando um desempenho inferior em termos de risco empírico.

5.2.4 Conjunto de dados MNIST

5.2.4.1 Desempenho na Curva AR em MNIST

A curva de Acurácia e Rejeição (AR) para o conjunto de dados MNIST é apresentada na Figura (10a) apresenta a variação da acurácia média e da taxa de rejeição média dos classificadores, enquanto a Figura (10b) mostra essas mesmas variações, mas com a aplicação de um fator de escala na reta da acurácia para melhor visualização dos resultados.

Figura 10 – Relação entre Taxas de Acurácia e Reieição - MNIST



Fonte: Autor.

Em termos de desempenho geral, o RO_VOTING se apresenta como o classificador mais eficaz, seguido de perto pelo RO_RF e RO_MLP que um certo ponto é ultrapassado pelo RO_GNB, como mostrado na Figura (10b). No entanto, o modelo RO_XGB apresentou o pior desempenho, indicando que ele tem mais dificuldade em manter um equilíbrio eficiente entre acurácia e rejeição.

Para proporcionar uma visão global do desempenho dos classificadores no conjunto de dados MNIST, a Tabela 20 apresenta os resultados com base em diferentes valores de w_r em todos os modelos.

Tabela 20 – Acurácia e Rejeição (ACC% e TR%) em MNIST

w_r	RO_XGB		RO_SVM		RO_KNN		RO_MLP		RO_GNB		RO_RF		RO_VOTING	
	ACC%	REJ%	ACC%	REJ%	ACC%	REJ%	ACC%	REJ%	ACC%	REJ%	ACC%	REJ%	ACC%	REJ%
0.08	99.95	16.78	99.94	31.47	99.29	6.56	99.58	11.00	81.42	6.22	100.00	83.58	99.90	16.18
0.15	99.78	12.14	99.90	19.12	99.29	6.56	99.46	8.10	80.77	4.71	100.00	70.60	99.87	11.24
0.22	99.60	9.96	99.87	12.94	99.29	6.56	99.29	6.28	80.44	3.72	100.00	60.29	99.85	8.76
0.30	99.36	8.26	99.82	9.03	99.29	6.56	99.04	4.72	80.07	2.86	100.00	48.78	99.76	6.93
0.38	99.21	6.76	99.76	6.43	99.29	6.47	98.86	3.78	79.85	2.26	100.00	38.14	99.63	5.51
0.45	98.97	5.50	99.72	5.11	98.28	1.49	98.63	2.94	79.40	1.36	100.00	30.53	99.40	3.50
0.52	98.47	3.86	99.64	4.14	97.60	0.43	98.16	1.65	79.10	0.10	100.00	23.71	99.09	2.32
0.60	97.99	2.50	99.53	2.96	97.60	0.43	97.71	0.61	79.05	0.01	99.95	16.53	98.90	1.47
0.68	97.46	1.18	99.28	1.50	97.43	0.00	97.54	0.21	79.04	0.00	99.67	10.18	98.59	0.50
0.75	96.98	0.33	99.06	0.64	97.43	0.00	97.42	0.06	79.04	0.00	99.17	5.08	98.44	0.21
0.82	96.80	0.07	98.79	0.04	97.43	0.00	97.37	0.00	79.04	0.00	97.96	1.12	98.15	0.01
0.90	96.75	0.00	98.76	0.00	97.43	0.00	97.37	0.00	79.04	0.00	97.39	0.00	98.11	0.00

5.2.4.2 Análise dos Classificadores em Relação ao Risco Empírico em MNIST

Para uma avaliação mais crítica dos classificadores, consideramos o valor do Risco Empírico (REMP) associado ao classificador de votação (RO_VOTING) para três diferentes configurações de w_r : 0.15, 0.44 e 0.81.

Tabela 21 – Rendimento dos classificadores para $w_r = 0.15$

Classificador	ACC (%)	REJ (%)	ERR (%)	REMP
RO_KNN	0.99	0.01	0.07	0.0170
RO_MLP	0.99	0.01	0.08	0.0176
RO_VOTING	1.00	0.00	0.11	0.0181
RO_XGB	1.00	0.00	0.12	0.0204
RO_SVM	1.00	0.00	0.19	0.0297
RO_RF	1.00	0.00	0.71	0.1059
RO_GNB	0.81	0.19	0.05	0.1993

Fonte: Autor

Os dados apresentados na Tabela 21 indicam que o classificador RO_VOTING possui um REMP de 0.0181, destacando-se por uma taxa de rejeição de 0%, mas apresentando erro de 11% o que o fez ficar em terceiro colocado. Os classificadores RO_KNN e RO_MLP apresentam REMP de 0.0170 e 0.0176, respectivamente, ambos com taxa de rejeição de 1% sendo os melhores para $w_r = 0.15$. O RO_XGB e o RO_SVM, com REMP de 0.0204 e 0.0297, também mostram taxa de rejeição de 0%. O RO_RF, com REMP de 0.1059 e taxa de rejeição de 0%, e o RO_GNB, com REMP de 0.1993 e taxa de rejeição de 19%, apresentam desempenho inferior. Essa análise sugere que o RO_VOTING se destaca entre os modelos avaliados, enquanto o RO_GNB requer melhorias significativas.

Tabela 22 – Rendimento dos classificadores para $w_r = 0.45$

Classificador	ACC (%)	ERR (%)	REJ (%)	REMP
RO_VOTING	0.99	0.01	0.04	0.0218
RO_SVM	1.00	0.00	0.05	0.0258
RO_MLP	0.99	0.01	0.03	0.0270
RO_KNN	0.98	0.02	0.01	0.0239
RO_XGB	0.99	0.01	0.06	0.0350
RO_RF	1.00	0.00	0.31	0.1374
RO_GNB	0.79	0.21	0.01	0.2121

Fonte: Autor

Para a configuração com $w_r = 0.45$, apresentada na Tabela 22, o classificador RO_VOTING apresenta um REMF de 0.0218, destacando-se por uma taxa de rejeição de 4% com o melhor desempenho para o risco empírico. O RO_MLP, com REMF de 0.0270 e taxa de rejeição de 3%, apresenta desempenho semelhante. O RO_SVM, que alcança uma acurácia de 100%, tem um REMF de 0.0258, tornando-se uma opção viável.

Os classificadores RO_XGB e RO_KNN apresentam REMF de 0.0350 e 0.0239, respectivamente, ambos com taxas de rejeição reduzidas. O RO_RF, com REMF de 0.1374 e taxa de rejeição de 31%, mostra desempenho inferior. O RO_GNB, com REMF de 0.2121 e taxa de rejeição de 21%, apresenta a maior ineficiência na minimização do risco empírico. Esta análise sugere que o RO_VOTING e o RO_SVM são os modelos mais eficazes na configuração avaliada.

Tabela 23 – Rendimento dos classificadores para $w_r = 0.82$

Classificador	ACC (%)	ERR (%)	REJ (%)	REMP
RO_SVM	0.99	0.01	0.00	0.0124
RO_VOTING	0.98	0.02	0.00	0.0186
RO_KNN	0.97	0.03	0.00	0.0257
RO_MLP	0.97	0.03	0.00	0.0263
RO_RF	0.98	0.02	0.01	0.0296
RO_XGB	0.97	0.03	0.00	0.0325
RO_GNB	0.79	0.21	0.00	0.2096

Fonte: Autor

Na Tabela 23, o classificador RO_SVM destaca-se com um REMF de 0.0124, apresentando a melhor performance entre os avaliados. O RO_VOTING, com REMF de 0.0186, segue em segundo lugar, enquanto o RO_KNN e o RO_MLP apresentam REMF de 0.0257 e 0.0263, respectivamente. Todos esses modelos demonstram uma forte capacidade de minimizar o risco empírico.

Por outro lado, os classificadores RO_RF e RO_XGB têm desempenhos inferiores, com REMF de 0.0296 e 0.0325, refletindo uma maior ineficiência. O RO_GNB, com

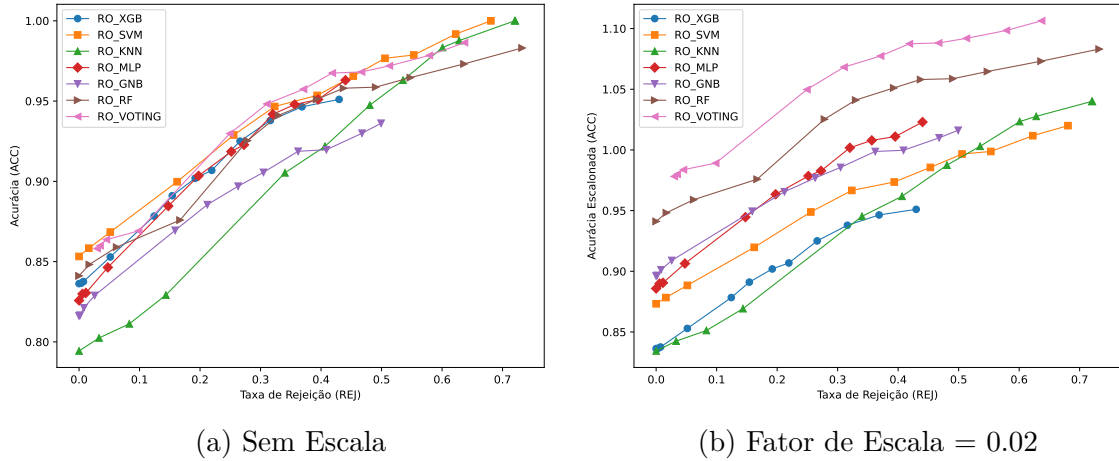
um REMP significativamente maior de 0.2096, evidencia o pior desempenho, indicando dificuldades na classificação. Esses resultados ressaltam a superioridade do RO_SVM e RO_VOTING em relação aos demais modelos na configuração analisada.

5.2.5 Conjunto de dados VCP

5.2.5.1 Desempenho na Curva AR em VCP

A curva de Acurácia e Rejeição (AR) para o conjunto de dados VCP é apresentada nas A Figura 11a apresenta a variação da acurácia média e da taxa de rejeição média dos classificadores, enquanto a Figura 11b mostra essas mesmas variações, mas com a aplicação de um fator de escala na reta da acurácia para melhor visualização dos resultados.

Figura 11 – Relação entre Taxas de Acurácia e Rejeição - VCP



(a) Sem Escala

(b) Fator de Escala = 0.02

Fonte: Autor.

A curva AR de todos os modelos segue o padrão esperado para o experimento assim como o experimento anterior, com exceção do comportamento observado no RO_VOTING já explicado nos resultados anteriores.

Em termos de desempenho geral, o RO_VOTING ainda se destaca como o classificador mais eficaz, seguido de perto pelo RO_RF, RO_MLP e RO_GNB, como mostrado na Figura 11b. No entanto, o modelo RO_KNN apresentou o pior desempenho na maior parte observável.

Para proporcionar uma visão global do desempenho dos classificadores no conjunto de dados VCP, a Tabela 24 apresenta os resultados com base em diferentes valores de w_r em todos os modelos.

Tabela 24 – Acurácia e Rejeição (ACC% e REJ%) em VCP

w_r	RO_XGB		RO_SVM		RO_KNN		RO_MLP		RO_GNB		RO_RF		RO_VOTING	
	ACC%	REJ%	ACC%	REJ%	ACC%	REJ%	ACC%	REJ%	ACC%	REJ%	ACC%	REJ%	ACC%	REJ%
0.06	95.11	42.98	100.0	68.06	100.0	72.10	96.30	44.03	93.62	49.92	98.31	73.23	98.64	63.71
0.11	94.64	36.85	99.17	62.26	100.0	72.02	95.10	39.52	93.00	46.77	97.31	63.63	97.85	57.90
0.17	93.79	31.61	97.87	55.32	98.77	62.82	94.79	35.65	91.97	40.89	96.46	54.76	97.20	51.29
0.22	92.50	26.61	97.66	50.56	98.34	60.08	94.18	32.02	91.87	36.21	95.87	49.03	96.81	46.69
0.28	90.68	21.94	96.56	45.32	96.30	53.55	92.28	27.26	90.56	30.48	95.80	43.71	96.75	41.85
0.33	90.19	19.19	95.36	39.35	94.75	48.06	91.85	25.16	89.70	26.29	95.11	39.27	95.74	37.10
0.39	89.11	15.40	94.66	32.34	92.18	40.65	90.35	19.76	88.55	21.21	94.11	32.98	94.82	31.05
0.44	87.84	12.42	92.89	25.56	90.53	34.03	88.46	14.76	86.95	15.89	92.53	27.82	92.98	24.92
0.50	85.29	5.16	89.97	16.21	82.92	14.35	84.64	4.76	82.89	2.58	87.58	16.69	86.91	9.84
0.56	83.76	0.73	86.84	5.16	81.12	8.31	83.05	1.13	82.12	0.81	85.90	6.21	86.37	4.52
0.61	83.65	0.32	85.84	1.61	80.25	3.31	82.99	0.56	81.66	0.16	84.82	1.69	86.00	3.47
0.67	83.63	0.00	85.32	0.00	79.44	0.00	82.58	0.00	81.61	0.00	84.11	0.00	85.81	2.98

Fonte: Autor

5.2.5.2 Análise dos Classificadores em Relação ao Risco Empírico em VCP

Para uma avaliação crítica dos classificadores, selecionamos as três tabelas mais relevantes com base no valor de Risco Empírico (REMP) associado ao classificador de votação (RO_VOTING). A seguir, apresentamos as análises para as configurações de $w_r = 0.11$, 0.39 e 0.67 .

Tabela 25 – Rendimento dos classificadores para $w_r = 0.11$

Classificador	ACC (%)	ERR (%)	REJ (%)	REMP
RO_SVM	99.17	0.83	62.26	0.0767
RO_KNN	100	0.00	72.02	0.0792
RO_VOTING	97.85	2.15	57.90	0.0852
RO_MLP	95.10	4.90	39.52	0.0925
RO_XGB	94.64	5.36	36.85	0.0941
RO_RF	97.31	2.69	63.63	0.0969
RO_GNB	93.00	7.00	46.77	0.1215

Fonte: Autor

A Tabela 25 para este valor de w_r os classificadores que melhor se saíram foram o RO_SVM e RO_KNN seguidos do RO_VOTING que está sendo penalizado em relação ao REMP devido a sua etapa subsequente de tomada de decisão em caso de empate, sendo exatamente a partir dele que começam a contabilizar taxas de rejeição. Os demais modelos seguem numa crescente evolução no valor do risco empírico.

Tabela 26 – Rendimento dos classificadores para $w_r = 0.39$

Classificador	ACC (%)	ERR (%)	REJ (%)	REMP
RO_XGB	89.11	10.89	15.40	0.1690
RO_VOTING	94.82	5.18	31.05	0.1729
RO_MLP	90.35	9.65	19.76	0.1736
RO_SVM	94.66	5.34	32.34	0.1795
RO_RF	94.11	5.89	32.98	0.1876
RO_GNB	88.55	11.45	21.21	0.1972
RO_KNN	92.18	7.82	40.65	0.2367

Fonte: Autor

A Tabela 26 apresenta os resultados do desempenho dos classificadores para $w_r = 0.39$. O classificador RO_VOTING, com um REMF de 0.1729, exibe um desempenho competitivo quando comparado aos outros modelos. Embora este valor seja ligeiramente inferior ao REMF do RO_XGB, que obteve 0.169, e praticamente equivalente ao do RO_SVM (0.179), o RO_KNN se destaca por apresentar uma taxa de erro de 40.65%. Já o RO_GNB por apresentar a maior taxa de rejeição (11.45%) desta amostra.

Tabela 27 – Rendimento dos classificadores para $w_r = 0.67$

Classificador	ACC (%)	ERR (%)	REJ (%)	REMP
RO_SVM	85.32	14.68	0.00	0.1468
RO_RF	84.11	15.89	0.00	0.1589
RO_VOTING	85.81	14.19	2.98	0.1619
RO_XGB	83.63	16.37	0.00	0.1637
RO_MLP	82.58	17.42	0.00	0.1742
RO_GNB	81.61	18.39	0.00	0.1839
RO_KNN	79.44	20.56	0.00	0.2056

Fonte: Autor

A Tabela 27 mostra que o classificador RO_VOTING, com um REMF de 0.1619, produz resultados semelhantes aos classificadores sem rejeição já que o único classificador que rejeita é o RO_VOTING devido a estratégia de rejeição para empates.

Os modelos RO_SVM e RO_RF, ambos com REMF de 0.1468 e 0.1589 respectivamente, apresentam ótimos resultados dentro do seu padrão sem rejeição. Apesar de evitarem a rejeição, apresentam resultados aproximados em comparação ao RO_VOTING.

Por outro lado, o RO_KNN apresenta o pior REMF de 0.206, apresentando uma alta taxa de erro (20.56%). Consequentemente levando a uma baixa acurácia, resultando em um desempenho geral inferior, o que destaca a limitação deste modelo em lidar com incertezas de forma eficaz.

Em conclusão, os classificadores RO_SVM e RO_XGB são consistentemente os melhores em termos de menor risco empírico para a maioria dos valores de w_r , com

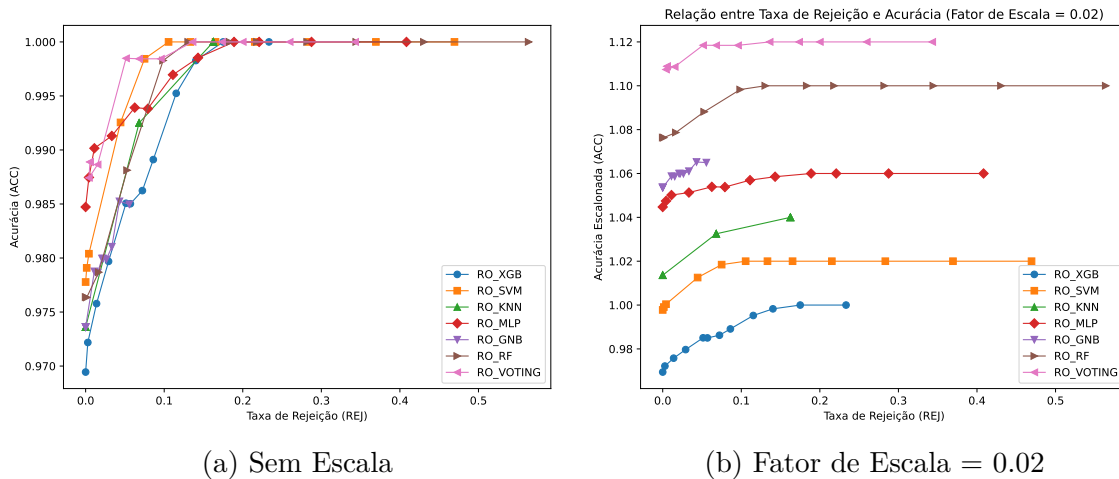
o RO_VOTING também mostrando um desempenho competitivo logo em seguida. O RO_KNN e o RO_GNB tendem a apresentar maior risco empírico, indicando que esses modelos podem ser menos adequados dependendo do valor de w_r para o conjunto de dados VCP.

5.2.6 Conjunto de dados WINE

5.2.6.1 Desempenho na Curva AR em WINE

A curva de Acurácia e Rejeição (AR) para o conjunto de dados WINE é apresentada na Figura 12a. A Figura 12a apresenta a variação da acurácia média e da taxa de rejeição média dos classificadores, enquanto a Figura 12b mostra essas mesmas variações, mas com a aplicação de um fator de escala na reta da acurácia para melhor visualização dos resultados.

Figura 12 – Relação entre Taxas de Acurácia e Rejeição - WINE



Fonte: Autor.

Em termos de desempenho geral, o RO_VOTING se destaca como o classificador mais eficaz, seguido de perto pelo RO_RF, RO_GNB e RO_MLP, como mostrado na Figura 12b. No entanto, o modelo RO_XGB apresentou o pior desempenho na maior parte observável.

Para proporcionar uma visão global do desempenho dos classificadores no conjunto de dados WINE, a Tabela 28 apresenta os resultados com base em diferentes valores de w_r em todos os modelos.

5.2.6.2 Análise dos Classificadores em Relação ao Risco Empírico em WINE

Para uma avaliação mais crítica dos classificadores, selecionamos as três tabelas mais relevantes com base no valor de Risco Empírico (REMP) associado ao classificador de votação (RO_VOTING). A seguir, apresentamos as análises para as configurações de $w_r = 0.11$, 0.39 e 0.67 .

Tabela 28 – Acurácia e Rejeição ($ACC \pm REJ$) em WINE

w_r	RO_XGB		RO_SVM		RO_KNN		RO_MLP		RO_GNB		RO_RF		RO_VOTING	
	ACC%	REJ%	ACC%	REJ%	ACC%	REJ%	ACC%	REJ%	ACC%	REJ%	ACC%	REJ%	ACC%	REJ%
0.06	100.0	23.33	100.0	46.94	100.0	16.25	100.0	40.83	98.50	5.56	100.0	56.39	100.0	34.31
0.11	100.0	17.50	100.0	36.94	100.0	16.25	100.0	28.75	98.52	4.31	100.0	43.06	100.0	25.97
0.17	99.83	14.03	100.0	28.33	100.0	16.25	100.0	22.08	98.10	3.33	100.0	34.44	100.0	20.00
0.22	99.52	11.53	100.0	21.53	100.0	16.25	100.0	18.89	97.99	2.64	100.0	28.19	100.0	17.36
0.28	98.91	8.61	100.0	16.53	100.0	16.25	99.85	14.31	98.00	2.22	100.0	21.81	100.0	13.61
0.33	98.62	7.22	100.0	13.33	99.25	6.81	99.70	11.11	98.00	2.08	100.0	18.33	99.84	9.58
0.39	98.50	5.69	100.0	10.56	99.25	6.81	99.38	7.92	97.86	1.53	100.0	13.06	99.84	6.81
0.44	98.51	5.14	99.84	7.50	99.25	6.81	99.39	6.25	97.88	1.11	99.83	9.86	99.85	5.14
0.50	97.97	2.92	99.25	4.44	97.36	0.00	99.13	3.33	97.36	0.00	98.81	5.28	98.87	1.53
0.56	97.58	1.39	98.04	0.42	97.36	0.00	99.02	1.11	97.36	0.00	97.87	1.67	98.74	0.42
0.61	97.22	0.28	97.91	0.14	97.36	0.00	98.75	0.42	97.36	0.00	97.63	0.14	98.89	0.56
0.67	96.94	0.00	97.78	0.00	97.36	0.00	98.47	0.00	97.36	0.00	97.64	0.00	98.89	0.56

Fonte: Autor

Tabela 29 – Rendimento dos classificadores para $w_r = 0.11$

Classificador	ACC (%)	ERR (%)	REJ (%)	REMP
RO_KNN	100.0	0.00	16.25	0.0179
RO_XGB	100.0	0.00	17.50	0.0192
RO_GNB	98.52	1.48	4.31	0.0195
RO_VOTING	100.0	0.00	25.97	0.0286
RO_MLP	100.0	0.00	28.75	0.0316
RO_SVM	100.0	0.00	36.94	0.0406
RO_RF	100.0	0.00	43.06	0.0474

Fonte: Autor

O conjunto de dados WINE apresenta padrões que possibilitam uma alta taxa de acerto (ACC). A Tabela 29 evidencia que o modelo RO_VOTING apresenta um REMF de 0.0286 apresentando seu pior resultado em relação aos outros. Embora este valor seja maior em comparação ao RO_KNN, que obteve um REMF de 0.0179, e ao RO_XGB, com 0.0192, o RO_VOTING ainda fica atrás até mesmo do RO_GNB. Destaca-se que, entre os classificadores analisados, o RO_RF foi o único a apresentar uma taxa de rejeição significativa de 43.06%. Neste grupo apenas o RO_GNB apresentou erro de classificação.

Tabela 30 – Rendimento dos classificadores para $w_r = 0.39$

Classificador	ACC (%)	ERR (%)	REJ (%)	REMP
RO_GNB	97.86	2.14	1.53	0.0273
RO_VOTING	99.84	0.16	6.81	0.0281
RO_KNN	99.25	0.75	6.81	0.0340
RO_MLP	99.38	0.62	7.92	0.0371
RO_XGB	98.50	1.50	5.69	0.0372
RO_SVM	100.0	0.00	10.56	0.0412
RO_RF	100.0	0.00	13.06	0.0509

Fonte: Autor

A Tabela 30 apresenta os resultados do desempenho dos classificadores para w_r

= 0.39. O classificador RO_VOTING, com um REMP de 0.0281, exibe um desempenho competitivo em comparação aos outros modelos. Embora este valor seja ligeiramente maior em relação ao REMP do RO_GNB, que obteve 0.0273, e menor que o do RO_KNN (0.0340), o RO_VOTING demonstra robustez, apesar de apresentar uma taxa de rejeição de 6.81%.

Já o classificador RO_RF ostenta a pior posição com um REMP de 0.0509 influenciado pela alta taxa de rejeição de 13.06%. A análise desses resultados reforça a importância de considerar tanto a acurácia quanto a rejeição na avaliação da eficácia dos modelos, especialmente quando se observa que a taxa de rejeição pode impactar o REMP significativamente.

Tabela 31 – Rendimento dos classificadores para $w_r = 0.67$

Classificador	ACC (%)	ERR (%)	REJ (%)	REMP
RO_VOTING	98.89	1.11	0.56	0.0148
RO_MLP	98.47	1.53	0.00	0.0153
RO_SVM	97.78	2.22	0.00	0.0222
RO_RF	97.64	2.36	0.00	0.0236
RO_KNN	97.36	2.64	0.00	0.0264
RO_GNB	97.36	2.64	0.00	0.0264
RO_XGB	96.94	3.06	0.00	0.0306

Fonte: Autor

A Tabela 31 mostra que o classificador RO_VOTING, com um REMP de 0.0148, apresenta o melhor desempenho entre os modelos listados. Com uma taxa de rejeição de 0.56%, o RO_VOTING consegue manter um REMP muito baixo.

Os demais resultados são comparados aos modelos sem opção de rejeição, já que este é o limiar máximo que para rejeição.

Em conclusão, os modelos que melhor performaram foram RO_KNN, RO_GNB e RO_VOTING. Por outro lado os modelos que apresentaram os piores resultados foram RO_RF e RO_XGB têm um risco empírico maior, indicando que podem ser menos adequados dependendo do valor de w_r no conjunto WINE.

5.3 Análise geral dos Resultados

5.3.1 Análise comportamento curva AR

Ao analisar o desempenho dos classificadores em relação à curva de Acurácia e Rejeição (AR), o modelo RO_VOTING consistentemente se destacou como o classificador mais eficaz, apresentando um equilíbrio superior entre acurácia e rejeição em todos os conjuntos de dados avaliados, seguido do RO_RF que esteve sempre com resultados

semelhantes ao RO_VOTING. Nas análises visuais, é possível identificar que os resultados do RO_VOTING superam os dos modelos individuais, confirmando a vantagem dos comitês de classificação neste quesito.

Entre os modelos individuais, o RO_MLP frequentemente demonstrou desempenho próximo ao do RO_VOTING, enquanto o RO_XGB apresentou resultados inferiores, especialmente na manutenção de uma alta taxa de acurácia à medida que a taxa de rejeição aumentava. Especificamente para o conjunto de dados VCP, o RO_KNN teve um desempenho menos eficiente.

5.3.2 Análise com base no risco empírico (REMP)

Analizamos o desempenho com base no Risco Empírico (REMP) em diferentes configurações de w_r . A análise é dividida por conjuntos de dados: BCW, VCP, IRIS e WINE, considerando os valores de w_r especificados.

Para o conjunto de dados BCW de modo geral o RO_VOTING se destaca por apresentar resultados de segundo e terceiro lugar, liderado pelos resultados do RO_MLP em seguida o RO_XGB. Na maioria dos casos o pior resultado foi do RO_GNB que aparece em último colocado em relação ao desempenho nos valores de risco empírico.

No conjunto de dados FISH o melhor desempenho foi para RO_MLP seguido pelo RO_VOTING e o RO_SVM que tiveram desempenho com valores aproximados de forma geral, desta vez o pior desempenho foi para o RO_RF seguido do RO_GNB.

Para o conjunto de dados IRIS modelo o RO_VOTING segue a tendência dos resultados anteriores sempre em segundo ou terceiro colocado, tendo o RO_MLP e o RO_SVM e o RO_KNN se destacando como melhores. O pior resultado foi variável.

Em relação ao conjunto MNIST o melhor modelo foi o RO_SVM e o RO_VOTING, com um resultado bom para o RO_KNN que teve uma melhora nos resultados para este conjunto. Novamente o RO_GNB se apresenta com o pior desempenho.

No conjunto de dados VCP o melhor desempenho foi para RO_SVM seguido pelo RO_VOTING e o RO_XGB que tiveram desempenho com valores aproximados de forma geral, desta vez o pior desempenho foi para o RO_KNN seguido do RO_GNB.

No conjunto de dados WINE o RO_VOTING apresentou variações de primeiro colocado a quarto colocado geral classificadores como o RO_XGB, RO_GNB e RO_KNN apresentaram resultados melhores do que em outros conjuntos de dados.

Os classificadores RO_SVM e RO_MLP consistentemente apresentam os menores riscos empíricos na maioria dos cenários, enquanto o RO_VOTING demonstra um desempenho competitivo na forma geral dos valores de w_r . Modelos como RO_XGB, RO_KNN e RO_GNB tendem a mostrar maior risco empírico, sugerindo que podem ser menos

eficazes para certos valores de w_r , foi possível observar uma grande variação entre todos os outros classificadores.

6 Conclusões e Trabalhos Futuros

Este capítulo final apresenta as conclusões, observações finais e identifica trabalhos futuros relacionados aos tópicos discutidos nesta dissertação.

6.1 Conclusões

Neste trabalho, apresentamos duas abordagens baseadas em classificação com opção de rejeição, uma aplicada a classificadores individuais e outra aplicada a comitês multi-classe, utilizando o índice de Gini como métrica de confiabilidade. A abordagem garante a eficácia para modelos individuais e apresenta resultados promissores quando aplicada a comitês de classificadores. Realizamos uma comparação entre modelos individuais e modelos em comitê usando quatro conjuntos de dados diferentes e seis modelos distintos, com foco na avaliação do modelo de voto de maioria simples (RO_VOTING). Os resultados indicaram que nosso modelo é capaz de generalizar os resultados e, em alguns casos, superar o desempenho dos modelos individuais quando observamos o desenvolvimento da curva AR, sempre se mostrando superior aos demais.

No entanto, a abordagem apresenta algumas limitações. Notamos que o modelo RO_VOTING tende a apresentar pontuações inferiores em termos de Risco Empírico, devido à etapa adicional necessária para garantir a confiança dos resultados em casos de empates durante a votação simples.

Os resultados das análises das curvas AR e do risco empírico (REMP) evidenciam a robustez do modelo RO_VOTING como uma abordagem efetiva para a classificação com rejeição, especialmente quando se busca um equilíbrio entre acurácia e rejeição. O RO_VOTING consistentemente demonstrou ser uma alternativa superior ou competitiva em comparação aos modelos individuais, validando a eficiência dos comitês de classificação na minimização do risco empírico e na manutenção de alta acurácia mesmo diante de uma crescente taxa de rejeição.

Nos conjuntos de dados BCW e IRIS, o RO_MLP se destacou como o classificador individual mais competitivo, frequentemente superando outros modelos como o RO_XGB, que mostrou variações de desempenho mais significativas, especialmente quando o w_r favoreceu uma maior penalização para erros. Em contrapartida, o RO_GNB apresentou os piores desempenhos nos cenários analisados, sendo consistentemente o classificador com maior risco empírico, reforçando sua menor robustez para aplicações que requerem alta precisão.

Para o conjunto de dados VCD, o RO_SVM emergiu como o classificador com

menor risco empírico, enquanto o RO_KNN teve dificuldades para manter competitividade, especialmente em contextos de alta rejeição. Este comportamento reflete uma tendência geral onde o RO_VOTING e modelos baseados em redes neurais (RO_MLP) lideram em situações que exigem um bom compromisso entre precisão e flexibilidade.

6.2 Trabalhos Futuros

Para trabalhos futuros, sugerimos a adaptação de uma função de desempate que evite penalizar ou simplesmente rejeitar resultados. Alternativamente, pode-se considerar o uso de métricas substitutas ou até mesmo a seleção de uma das classificações dos componentes-base mais confiáveis para aprimorar a estratégia de rejeição.

Além disso, a abordagem para identificar automaticamente o risco empírico e determinar o melhor classificador dentro dos intervalos de limiares pretendidos deve ser explorada. Essa abordagem pode fornecer suporte decisivo na tomada de decisão e na resolução de problemas.

Finalmente, a integração de estratégias de rejeição em outros modelos de ensemble, como Boosting, pode abrir novas possibilidades para aprimorar a robustez e a precisão dos sistemas de classificação em conjunto.

Referências

ALGEHYNE, E. A. et al. Fuzzy Neural Network Expert System with an Improved Gini Index Random Forest-Based Feature Importance Measure Algorithm for Early Diagnosis of Breast Cancer in Saudi Arabia. *Big Data and Cognitive Computing*, v. 6, n. 1, p. 13, mar. 2022. ISSN 2504-2289. Number: 1 Publisher: Multidisciplinary Digital Publishing Institute. Disponível em: <<https://www.mdpi.com/2504-2289/6/1/13>>. Citado na página 49.

ALMEIDA, E. D. Classificação Ordinal com Opção de Rejeição. 2010. Citado 2 vezes nas páginas 62 e 63.

ALMEIDA, W. D. d. *Detecção de erros em canais de comunicação via classificadores de reconhecimento de padrão com opção de rejeição*. Tese (Dissertação) — Instituto Federal do Ceará, 2021. Citado 3 vezes nas páginas 59, 63 e 65.

BALLABIO, D.; TODESCHINI, R.; CONSONNI, V. Chapter 5 - Recent Advances in High-Level Fusion Methods to Classify Multiple Analytical Chemical Data. In: COCCHI, M. (Ed.). *Data Fusion Methodology and Applications*. [S.l.]: Elsevier, 2019, (Data Handling in Science and Technology, v. 31). p. 129–155. ISSN: 0922-3487. Citado na página 44.

BISHOP, C. *Pattern Recognition and Machine Learning*. Springer New York, 2016. (Information Science and Statistics). ISBN 978-1-4939-3843-8. Disponível em: <<https://books.google.com.br/books?id=kOXDtAEACAAJ>>. Citado 4 vezes nas páginas 29, 35, 43 e 49.

BOSER, B. E.; GUYON, I. M.; VAPNIK, V. N. A training algorithm for optimal margin classifiers. In: *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*. New York, NY, USA: Association for Computing Machinery, 1992. (COLT '92), p. 144–152. ISBN 089791497X. Disponível em: <<https://doi.org/10.1145/130385.130401>>. Citado na página 41.

BOUNSIAR, A.; BEAUSEROY, P.; GRALL-MAËS, E. General solution and learning method for binary classification with performance constraints. *Pattern Recognition Letters*, v. 29, n. 10, p. 1455–1465, jul. 2008. ISSN 0167-8655. Citado na página 63.

BREIMAN, L. Random Forests. *Machine Learning*, v. 45, n. 1, p. 5–32, out. 2001. ISSN 1573-0565. Disponível em: <<https://doi.org/10.1023/A:1010933404324>>. Citado 2 vezes nas páginas 48 e 49.

CAIE, P. D.; DIMITRIOU, N.; ARANDJELOVIĆ, O. Chapter 8 - Precision medicine in digital pathology via image analysis and machine learning. In: COHEN, S. (Ed.). *Artificial Intelligence and Deep Learning in Pathology*. Elsevier, 2021. p. 149–173. ISBN 978-0-323-67538-3. Disponível em: <<https://www.sciencedirect.com/science/article/pii/B9780323675383000087>>. Citado na página 48.

CHEN, T.; GUESTRIN, C. Xgboost: A scalable tree boosting system. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data*

Mining. New York, NY, USA: Association for Computing Machinery, 2016. (KDD '16), p. 785–794. ISBN 9781450342322. Citado na página 46.

CHEN, T.; GUESTRIN, C. Xgboost: A scalable tree boosting system. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, NY, USA: Association for Computing Machinery, 2016. (KDD '16), p. 785–794. ISBN 9781450342322. Disponível em: <<https://doi.org/10.1145/2939672.2939785>>. Citado na página 47.

CHOW, C. On optimum recognition error and reject tradeoff. *IEEE Transactions on Information Theory*, v. 16, n. 1, p. 41–46, 1970. Citado 3 vezes nas páginas 31, 51 e 58.

CHOW, C. K. An optimum character recognition system using decision functions. *IRE Transactions on Electronic Computers*, EC-6, n. 4, p. 247–254, 1957. Citado 5 vezes nas páginas 29, 58, 59, 60 e 61.

CORDELLA, L. P. et al. Classification reliability and its use in multi-classifier systems. *Lecture Notes in Computer Science*, 1997. Citado na página 31.

CORDELLA, L. P. et al. A method for improving classification reliability of multilayer perceptrons. *IEEE transactions on neural networks*, v. 6, n. 5, p. 1140–1147, 1995. ISSN 1045-9227. Citado na página 54.

CORTES, C.; VAPNIK, V. Support-vector networks. *Machine Learning*, v. 20, n. 3, p. 273–297, set. 1995. ISSN 1573-0565. Citado na página 40.

COVER, T.; HART, P. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, v. 13, n. 1, p. 21–27, 1967. Citado na página 41.

DOGAN, A.; BIRANT, D. A Weighted Majority Voting Ensemble Approach for Classification. In: *2019 4th International Conference on Computer Science and Engineering (UBMK)*. [S.l.: s.n.], 2019. p. 1–6. Citado na página 45.

DUBUISSON, B.; MASSON, M. A statistical decision rule with incomplete knowledge about classes. *Pattern Recognition*, v. 26, n. 1, p. 155–165, 1993. ISSN 0031-3203. Citado 2 vezes nas páginas 31 e 52.

ENGELLEN, J. E. van; HOOS, H. H. A survey on semi-supervised learning. *Machine Learning*, v. 109, n. 2, p. 373–440, fev. 2020. ISSN 1573-0565. Disponível em: <<https://link.springer.com/article/10.1007/s10994-019-05855-6>>. Citado na página 36.

FERREIRA, A. J.; FIGUEIREDO, M. A. T. Boosting Algorithms: A Review of Methods, Theory, and Applications. In: ZHANG, C.; MA, Y. (Ed.). *Ensemble Machine Learning: Methods and Applications*. New York, NY: Springer, 2012. p. 35–85. ISBN 978-1-4419-9326-7. Citado na página 46.

FIX, E.; HODGES, J. L. Discriminatory Analysis. Nonparametric Discrimination: Consistency Properties. *International Statistical Review / Revue Internationale de Statistique*, v. 57, n. 3, p. 238–247, 1989. ISSN 0306-7734. Publisher: [Wiley, International Statistical Institute (ISI)]. Disponível em: <<https://www.jstor.org/stable/1403797>>. Citado na página 41.

- FREUND, Y.; SCHAPIRE, R.; ABE, N. A short introduction to boosting. *Journal-Japanese Society For Artificial Intelligence*, JAPANESE SOC ARTIFICIAL INTELL, v. 14, n. 771-780, p. 1612, 1999. Citado na página 46.
- FUKUNAGA, K. *Introduction to statistical pattern recognition*. [S.l.]: Elsevier, 1990. Citado 2 vezes nas páginas 31 e 50.
- FUMERA, G.; ROLI, F. Support Vector Machines with Embedded Reject Option. In: LEE, S.-W.; VERRI, A. (Ed.). *Pattern Recognition with Support Vector Machines*. Berlin, Heidelberg: Springer, 2002. (Lecture Notes in Computer Science), p. 68–82. ISBN 978-3-540-45665-0. Citado na página 63.
- GANAI, M. A. et al. Ensemble deep learning: A review. *Engineering Applications of Artificial Intelligence*, 2021. Citado na página 32.
- GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep Learning*. MIT Press, 2016. (Adaptive Computation and Machine Learning series). ISBN 978-0-262-03561-3. Disponível em: <<https://books.google.com.br/books?id=Np9SDQAAQBAJ>>. Citado na página 29.
- HASAN, M. K.; JAWAD, M. T. Breast cancer classification using ensemble of machine learning boosting algorithms. In: *2022 International Conference on Inventive Computation Technologies (ICICT)*. [S.l.: s.n.], 2022. p. 444–451. Citado na página 30.
- HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. Overview of Supervised Learning. In: HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. (Ed.). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. New York, NY: Springer, 2009. p. 9–41. ISBN 978-0-387-84858-7. Disponível em: <https://doi.org/10.1007/978-0-387-84858-7_2>. Citado na página 29.
- HAYES, A. *Bayes Theorem : a Quick-Start Beginner's Guide*. CreateSpace Independent Publishing Platform, 2017. ISBN 9781542493598. Disponível em: <<https://books.google.com.br/books?id=QKdYMQAACAAJ>>. Citado na página 52.
- HELLMAN, M. E. The nearest neighbor classification rule with a reject option. *IEEE Transactions on Systems Science and Cybernetics*, v. 6, n. 3, p. 179–185, 1970. Citado 2 vezes nas páginas 31 e 52.
- HENDRICKX, K. et al. Machine learning with a reject option: A survey. *ArXiv*, 2021. Citado 2 vezes nas páginas 31 e 57.
- HENDRYCKS, D.; GIMPEL, K. *A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks*. arXiv, 2018. ArXiv:1610.02136 [cs]. Disponível em: <<http://arxiv.org/abs/1610.02136>>. Citado 2 vezes nas páginas 53 e 54.
- KHATIBISEPEHR, S.; HUANG, B. Bayesian methods for process identification with outliers. In: *2012 American Control Conference (ACC)*. [S.l.: s.n.], 2012. p. 3516–3521. Citado na página 52.
- KUAN, X. How has the literature on gini's index evolved in the past 80 years. *China Economic Quarterly*, 2003. Citado na página 30.

- KUNCHEVA, L. I. On the optimality of Naïve Bayes with dependent binary features. *Pattern Recognition Letters*, v. 27, n. 7, p. 830–837, maio 2006. ISSN 0167-8655. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0167865505003582>>. Citado na página 43.
- LAROUÏ, S. et al. *How to define a rejection class based on model learning*. 2021. Citado 2 vezes nas páginas 30 e 36.
- LEE, P. M. *Bayesian Statistics: An Introduction*. 4th. ed. [S.l.]: Wiley Publishing, 2012. ISBN 1118332571. Citado na página 52.
- MANTHEY, B.; RÖGLIN, H. Improved smoothed analysis of the k-means method. In: *Proceedings of the twentieth annual ACM-SIAM symposium on Discrete algorithms*. [S.l.]: Society for Industrial and Applied Mathematics, 2009. p. 461–470. Citado na página 36.
- MENA, J.; PUJOL, O.; VITRIÀ, J. Uncertainty-based rejection wrappers for black-box classifiers. *IEEE Access*, v. 8, p. 101721–101746, 2020. Citado na página 30.
- MIENYE, I. D.; SUN, Y. A survey of ensemble learning: Concepts, algorithms, applications, and prospects. *IEEE Access*, v. 10, p. 99129–99149, 2022. Citado na página 31.
- MIENYE, I. D.; SUN, Y. A Survey of Ensemble Learning: Concepts, Algorithms, Applications, and Prospects. *IEEE Access*, v. 10, p. 99129–99149, 2022. Citado 2 vezes nas páginas 46 e 48.
- MURPHY, K. P. *Machine Learning: A Probabilistic Perspective*. [S.l.]: The MIT Press, 2012. ISBN 0262018020. Citado na página 29.
- MÜLLER, J. et al. Finding competence regions in domain generalization. *ArXiv*, 2023. Citado na página 31.
- OLIVEIRA, A. C. d. *Máquina de aprendizagem mínima com opção de rejeição*. Tese (Dissertação) — Universidade Federal do Ceará, 2016. Citado 2 vezes nas páginas 60 e 65.
- OSAMOR, V. C.; OKEZIE, A. F. Enhancing the weighted voting ensemble algorithm for tuberculosis predictive diagnosis. *Scientific Reports*, v. 11, n. 1, p. 14806, jul. 2021. ISSN 2045-2322. Number: 1 Publisher: Nature Publishing Group. Citado na página 45.
- PLATT, J. Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods. *Adv. Large Margin Classif.*, v. 10, jun. 2000. Citado na página 53.
- ROCHA NETO, A. R. *SINPATCO II: Novas Estratégias de Aprendizado de Máquina para Classificação de Patologias da Coluna Vertebral*. Tese (Tese de Doutorado) — Universidade Federal do Ceará, 2011. Citado 6 vezes nas páginas 54, 57, 58, 60, 61 e 62.
- ROCHA NETO, A. R. et al. Diagnostic of pathology on the vertebral column with embedded reject option. In: *Pattern Recognition and Image Analysis*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011. p. 588–595. ISBN 978-3-642-21257-4. Citado 3 vezes nas páginas 30, 41 e 57.

- RUSSELL, S.; NORVIG, P. *Artificial Intelligence, Global Edition A Modern Approach*. Pearson Deutschland, 2021. 1168 p. ISBN 9781292401133. Disponível em: <<https://elibrary.pearson.de/book/99.150005/9781292401171>>. Citado na página 29.
- SOUSA, R.; MORA, B.; CARDOSO, J. S. An ordinal data method for the classification with reject option. In: *2009 International Conference on Machine Learning and Applications*. [S.l.: s.n.], 2009. p. 746–750. Citado 4 vezes nas páginas 29, 30, 58 e 63.
- SUTHAHARAN, S. Chapter 6 - A Cognitive Random Forest: An Intra- and Intercognitive Computing for Big Data Classification Under Cune Condition. In: GUDIVADA, V. N. et al. (Ed.). *Handbook of Statistics*. Elsevier, 2016. (Cognitive Computing: Theory and Applications, v. 35), p. 207–227. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0169716116300475>>. Citado na página 48.
- TAN, P. et al. *Introduction to Data Mining*. [S.l.]: Pearson, 2019. (What's New in Computer Science Series). ISBN 978-0-13-312890-1. Citado 7 vezes nas páginas 36, 41, 42, 44, 45, 46 e 48.
- THEODORIDIS, S. et al. *Introduction to pattern recognition: a matlab approach*. [S.l.]: Academic Press, 2010. Citado na página 35.
- VILLON, S. et al. A new method to control error rates in automated species identification with deep learning algorithms. *Scientific Reports*, 2020. Citado na página 32.
- WEINBERGER, K. Q.; SAUL, L. K. Distance metric learning for large margin nearest neighbor classification. *Journal of machine learning research*, v. 10, n. 2, 2009. Citado na página 52.
- YUAN, Y.; WU, L.; ZHANG, X. Gini-impurity index analysis. *IEEE Transactions on Information Forensics and Security*, 2021. Citado 2 vezes nas páginas 30 e 54.
- ZHANG, B.; ZHANG, Y.; JIANG, X. Feature selection for global tropospheric ozone prediction based on the BO-XGBoost-RFE algorithm. *Scientific Reports*, v. 12, p. 9244, jun. 2022. ISSN 2045-2322. Number: 1 Publisher: Nature Publishing Group. Citado 2 vezes nas páginas 47 e 48.
- ZHANG, H. The Optimality of Naive Bayes. *AAAI*, v. 2, jan. 2004. Disponível em: <<https://aaai.org/papers/flairs-2004-097/>>. Citado na página 43.
- ZHANG, X.-Y. et al. A survey on learning to reject. *Proceedings of the IEEE*, v. 111, n. 2, p. 185–215, 2023. Citado 2 vezes nas páginas 29 e 30.