Avaliação da influência de métodos de redução de dimensionalidade no desempenho de modelos de classificação

Mirelli de Castro Cesário
Instituto de Engenharia de Produção e Gestão
Universidade Federal de Itajubá
Itajubá, Brasil
mirellicesario12@gmail.com

Leonardo Lourenço de Souza
Instituto de Engenharia de Produção e Gestão
Universidade Federal de Itajubá
Itajubá, Brasil
leo.lourenco93@gmail.com

Matheus Costa Pereira
Instituto de Engenharia de Produção e Gestão
Universidade Federal de Itajubá
Itajubá, Brasil
matheusc_pereira@hotmail.com

Juliana Helena Daroz Gaudêncio
Instituto de Engenharia de Produção e Gestão
Universidade Federal de Itajubá
Itajubá, Brasil
juliana.gaudencio@unifei.edu.br

Matheus Brendon Francisco
Instituto de Engenharia de Produção e Gestão
Universidade Federal de Itajubá
Itajubá, Brasil
matheus_brendon@yahoo.com.br

Anderson Paulo de Paiva
Instituto de Engenharia de Produção e Gestão
Universidade Federal de Itajubá
Itajubá, Brasil
andersonppaiva@unifei.edu.br

***Resumo* - Este estudo investiga o impacto da seleção de variáveis e da redução de dimensionalidade no desempenho de modelos de classificação, por meio da aplicação da análise de componentes principais e da análise fatorial. Foram avaliadas seis bases de dados com diferentes características, comparando o desempenho dos modelos de *machine learning* nos distintos conjuntos e analisando o impacto do uso do *Grid Search*. Os resultados mostraram que as duas técnicas adotadas permitiram reduzir significativamente a dimensionalidade, com desempenho muito próximo ao obtido com as bases completas. A análise do tempo de processamento evidenciou o custo computacional envolvido, especialmente em modelos mais complexos, como o *Light Gradient Boosting Machine*. Conclui-se que a combinação entre redução de dimensionalidade, seleção de variáveis e otimização de hiperparâmetros contribui significativamente para a escolha do modelo mais adequado ao problema.**

Palavras-chave – *machine learning*, classificação, redução de dimensionalidade, análise de componentes principais, análise fatorial, custo computacional

I. INTRODUÇÃO

O *Machine Learning* (ML) é uma abordagem de análise de dados que automatiza o desenvolvimento de modelos preditivos, permitindo que algoritmos aprendam a partir da experiência e melhorem seu desempenho ao longo do tempo [1]. Entre os tipos de aprendizado, destaca-se a classificação, uma técnica de aprendizado supervisionado utilizada para prever a classe de novas observações com base em um conjunto de características preditivas [2-3] . Esse modelo tem sido muito explorado por identificar padrões sutis entre classes, os quais não são detectados por abordagens estatísticas tradicionais [4]. Ao serem treinados com conjuntos de dados representativos, os algoritmos de classificação buscam determinar a qual categoria pertencem os novos dados sem terem sido treinados previamente com esses dados [5], e reconhecer padrões e tendências presentes nos dados.

Os algoritmos de classificação são muito utilizados na ciência de dados para identificar padrões e categorizar resultados por meio de dados previamente rotuladas [6]. Com o avanço das tecnologias da informação e a crescente disponibilidade de dados, os modelos de classificação são aplicados em diversas áreas. Por exemplo, na saúde, são utilizados para auxiliar no diagnóstico automatizado de doenças, como a classificação de tipos de glaucoma com base em imagens de retina [7], e na análise de registros clínicos para suporte ao diagnóstico e prognóstico de pacientes com câncer, utilizando informações não estruturadas dos prontuários [1]. Na neurociência, são utilizados para decodificar estímulos cognitivos a partir de sinais eletroencefalográficos, permitindo distinguir diferentes padrões de percepção visual ou emocional [4]. Além disso, modelos de classificação são aplicados na detecção de anomalias geoquímicas associadas a mineralização [8], na previsão de comportamentos de usuários de *smartphones* com base em dados contextuais [3] e na previsão da geração de energia solar fotovoltaica a partir da classificação de padrões climáticos [9]. Também, nas telecomunicações para resolver problema de roteamento em redes ópticas elásticas [5], e em finanças para avaliação de risco de crédito a partir de dados demográficos e históricos financeiros [10].

Há uma ampla gama de algoritmos de classificação que podem ser utilizados em problemas de aprendizado supervisionado [2] como *Logistic Regression* (LR), *Decision Tree* (DT), *Random Forest* (RF), *K-Nearest Neighbors* (KNN), entre outros. Apesar de existirem inúmeros algoritmos, eles se baseiam em concepções diferentes, podendo ser provenientes de distâncias, árvores, *boosting*, redes neurais etc. Com isso, é importante avaliar cuidadosamente o desempenho, pois, a escolha do modelo é uma importante etapa do processo de tomada de decisão e impacta diretamente no resultado obtido [11].

A complexidade computacional é um fator determinante na escolha de algoritmos de ML, especialmente em aplicações com grandes volumes de dados. Gzar et al. [12] destacam que o tempo de execução e o uso de memória impactam diretamente a viabilidade do uso de modelos mais sofisticados em cenários reais. Complementarmente, Ziolkowski [13] ressalta a importância de equilibrar desempenho preditivo com o risco de *overfitting* e o custo de treinamento, recomendando o uso de regularização, validação cruzada e ajuste de hiperparâmetros como estratégias para mitigar esses efeitos. De maneira similar, Maree e Shehada [14] observam que modelos mais avançados, embora mais custosos em

termos computacionais, tendem a gerar resultados mais robustos em tarefas complexas.

Com o aumento da complexidade e da dimensionalidade das bases de dados, surgem alguns desafios, tais como redundância de informações, aumento no tempo de processamento e dificuldades na generalização dos modelos. O pré-processamento dos dados pode consumir de 50% a 80% do tempo total de desenvolvimento de um modelo de classificação [15]. Além disso, a complexidade do modelo pode impactar de forma significativa no tempo de processamento [16].

Técnicas de redução de dimensionalidade podem ser aplicadas com o objetivo de simplificar os dados, preservando as informações mais relevantes para a tarefa preditiva. Essa abordagem busca explorar as informações importantes contidas nos dados e eliminar ruídos, contribuindo para a melhoria do desempenho dos modelos e a prevenção de *overfitting* [17-18].

O pré-processamento de dados é uma etapa fundamental no processo de modelagem de ML [11]. A redução de dimensionalidade está presente nela, e podem ser utilizadas técnicas como: a análise de componentes principais (PCA) e a análise fatorial (FA), que possibilitam transformar o espaço original de variáveis em novos componentes ou fatores que preservam a variância ou estrutura latente dos dados. Esse processo reduz a dimensionalidade do problema e remove termos redundantes, o que contribui para diminuir o tempo de processamento e a complexidade do modelo [19].

Apesar do amplo uso de técnicas de redução de dimensionalidade, ainda existem lacunas quanto à compreensão de seus efeitos reais sobre o desempenho de modelos de classificação, especialmente no que se refere à eficiência e ao tempo de execução. Diante desse cenário, este estudo tem como objetivo comparar o desempenho de diferentes algoritmos de classificação aplicados a conjuntos de dados em três formatos: completos (original), com redução de dimensionalidade por PCA e por FA. Para isso, serão utilizadas bases de dados do repositório *UCI Machine Learning*, avaliando-se o tempo de processamento e o desempenho de cada modelo em cada cenário. Essa abordagem busca avaliar o impacto da dimensionalidade tanto nas métricas de desempenho quanto no custo computacional envolvido.

A metodologia proposta foi implementada em seis conjuntos de dados provenientes de áreas distintas. Cada conjunto possui quantidades diferentes de atributos, além de volumes variados de observações e classes. Avaliou-se sete algoritmos de aprendizado de máquina, que representam diferentes arquiteturas: KNN, LR, *Linear Discriminant Analysis* (LDA), RF, *XGBoost*, *LightGBM* e *Multi-Layer Perceptron* (MLP). A avaliação desses modelos foi conduzida com base nas métricas de acurácia, precisão, *recall* e F1-Score, complementada pela análise do tempo de processamento de cada algoritmo.

Os modelos com melhor desempenho, identificados a partir das combinações de fatores otimizadas, foram submetidos a um processo de otimização de hiperparâmetros utilizando o método *Grid Search*.

II. MATERIAIS E MÉTODOS

O trabalho foi desenvolvido em um ambiente de programação em *Python* e organizado em etapas: leitura e pré-processamento dos dados, criação dos *datasets* com a dimensionalidade reduzida, treinamento, teste e validação cruzada dos modelos, otimização de hiperparâmetros e, por fim, comparação dos resultados. De forma resumida, o fluxo pode ser visualizado na Fig. 1.

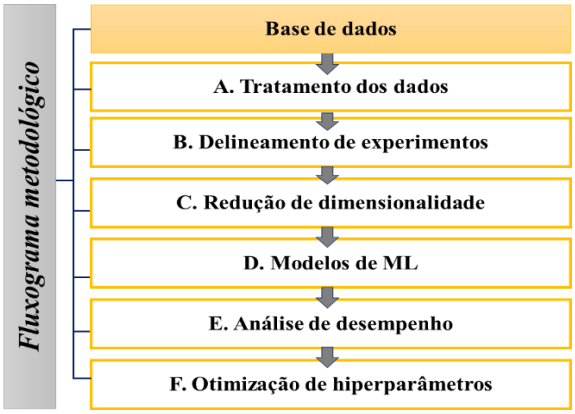

Fig. 1. Fluxograma metodológico

Inicialmente, foram definidos os *datasets a* serem utilizados. Nesse trabalho o foco são problemas de classificação, portanto, são utilizadas bases de dados destinadas a esse tipo de problema. Para avaliar cenários e dimensionalidades diferentes, foram consideradas as seguintes bases:

1) Rice [20]*:* Contém 3810 imagens de grãos de duas variedades de arroz, coletadas por meio de um sistema de visão computacional. Para cada grão, foram extraídas 7 características morfológicas;

2) Drybean [21]*:* Compreende 13611 imagens de grãos de 7 variedades de feijão seco, onde foram extraídas 16 características morfológicas;

3) Chemical [22]*:* Formada por dados de composição química de 40 fragmentos cerâmicos (corpo e esmalte) obtidos por fluorescência de raios-X por dispersão de energia. As amostras são provenientes dos fornos *Longquan* e *Jingdezhen*, abrangendo quatro eras culturais;

4) Raisin [23]*:* Possui 900 imagens de grãos de duas variedades de uva-passa, em proporções iguais. Extraiu-se 7 características morfológicas;

5) ForestFire [24]*:* Apresenta dados de duas regiões da Argélia, contendo 11 atributos e 244 instância classificadas em incêndio e não incêndio;

6) Vehicle Silhouettes [25]*:* Composta por informações de veículos para classificar a silhueta considerando quatro tipos distintos de veículos (ônibus de dois andares, van, Saab 9000 e Opel Manta 400), com base em 18 variáveis preditoras extraídas das imagens das silhuetas.

A. Tratamento prévio dos dados

No tratamento dos dados, a primeira etapa consiste em padronizá-los. Desse modo, as variáveis são normalizadas, de modo a apresentarem média igual a zero e desvio padrão igual a um. Esse procedimento visa minimizar problemas decorrentes da escala dos dados, que podem comprometer o

desempenho de determinados modelos estatísticos ou de ML especialmente nos casos sensíveis à magnitude das variáveis. Com os dados devidamente padronizados e os conjuntos de dados construídos, procede-se à etapa de definição dos delineamentos experimentais.

B. Delineamento dos experimentos

A metodologia de delineamento de experimentos utilizada nesse trabalho foi o fatorial de dois níveis. O delineamento fatorial contempla todas as combinações possíveis entre os níveis dos fatores experimentais e cada iteração do delineamento é representada, de forma codificada, pelos valores -1 ou +1. A codificação em seu nível baixo indica a exclusão do fator no modelo correspondente àquela iteração, enquanto o valor em seu nível alto sinaliza a sua inclusão.

C. Redução de dimensionalidade

Para a etapa de redução de dimensionalidade, foram adotadas duas abordagens distintas: PCA e FA. Inicialmente, os modelos foram avaliados utilizando a base de dados completa, sem a aplicação de técnicas de redução de dimensão, a fim de servir como referência comparativa. Posteriormente, a definição do número adequado de componentes principais foi baseada no critério de explicação acumulada da variância, considerando-se um valor mínimo de 85% da variância total dos dados. Para a FA, utilizou-se o mesmo número de fatores sugerido pela PCA, aplicando-se, adicionalmente, a rotação ortogonal do tipo *Varimax*.

A rotação *Varimax* tem como propósito tornar a estrutura fatorial mais interpretável, promovendo a simplificação das cargas fatoriais. Essa técnica maximiza a variância das cargas dentro de cada fator, favorecendo a formação de padrões em que cada variável apresenta alta correlação com apenas um fator, enquanto mantém correlações reduzidas com os demais. Como resultado, os fatores tendem a ser representados por subconjuntos distintos de variáveis, ortogonais e facilita a interpretação conceitual.

D. Modelos de Machine Learning

Nesta seção, são apresentados e descritos os algoritmos de ML empregados na investigação, abrangendo um total de sete métodos selecionados baseado em sua ampla utilização na literatura e em sua diversidade de abordagens para a modelagem preditiva, são eles:

- *K-Nearest Neighbors* (KNN): Classifica novas instâncias com base na classe majoritária de seus *k* vizinhos mais próximos no espaço de características, fundamentando-se na similaridade de distância;
- *Logistic Regression* (LR): É um modelo linear generalizado que estima a probabilidade de uma ocorrência binária, aplicando uma função sigmoide para mapear a combinação linear das características de entrada a um valor entre 0 e 1;
- *Linear Discriminant Analysis* (LDA): Busca otimizar a separação entre classes ao projetar os dados em um espaço de menor dimensionalidade, onde a razão entre a variância entre classes e a variância intraclasse é maximizada;
- *Random Forest* (RF): É um método de *ensemble* que agrega as previsões de múltiplas árvores de decisão construídas a partir de subconjuntos aleatórios de dados e características, reduzindo o *overfitting* e aumentando a robustez do modelo;
- *Extreme Gradient Boosting* (*XGBoost*): É uma implementação otimizada do *gradient boosting*, caracterizada por sua alta performance e escalabilidade, incorporando regularização para evitar *overfitting* e paralelização para eficiência computacional;
- *Light Gradient Boosting Machine* (*LightGBM*): É um framework rápido e eficiente para *gradient boosting*, projetado para alto desempenho. Ele usa um método de crescimento de árvores baseado em histogramas, além de técnicas como amostragem de gradientes e exclusão de variáveis menos importantes, o que acelera o treinamento e reduz o uso de memória;
- *Multi-Layer Perceptron* (MLP): É uma classe de redes neurais *feedforward* que consiste em múltiplas camadas de neurônios interconectados, incluindo pelo menos uma camada oculta, capaz de modelar relações não-lineares complexas nos dados.

As arquiteturas de redes neurais convolucionais (CNN) e recorrentes (RNN), embora amplamente utilizadas em tarefas de classificação, não foram consideradas neste estudo. Isso se deve ao fato de as bases utilizadas, apesar de derivadas de imagens ou sensores, estarem representadas em formato tabular, compostas por atributos numéricos previamente extraídos. Dessa forma, não apresentam estrutura sequencial, como em séries temporais, nem espacial, como em imagens, que são os contextos típicos de aplicação dessas redes. Assim, a escolha metodológica buscou garantir coerência com a natureza dos dados e com os objetivos centrais da pesquisa.

E. Análise de desempenho

Para avaliar o desempenho dos modelos de ML, foram utilizados quatro métricas: acurácia, precisão, *recall* e *F1-Score*.

- Acurácia (A): Representa a proporção de previsões corretas (verdadeiros positivos e verdadeiros negativos) em relação ao total de instâncias, sendo uma métrica geral da performance do modelo;
- *Recall* (R): Mede a proporção de verdadeiros positivos que foram corretamente identificados dentre todas as instâncias realmente positivas, refletindo a capacidade do modelo de evitar falsos negativos;
- Precisão (P): Quantifica a proporção de verdadeiros positivos entre todas as instâncias classificadas como positivas, indicando a capacidade do modelo de evitar falsos positivos;
- *F1-Score* (F1): É a média harmônica da precisão e do *recall*, oferecendo uma métrica balanceada da performance do modelo, especialmente útil em cenários com classes desbalanceadas.

Na sequência, utiliza-se a técnica de validação cruzada *Stratified K-Fold*, com cinco subdivisões (*folds*), em que, a cada rodada, uma parte dos dados é reservada para teste, enquanto o restante é utilizado para o treinamento. Diferentemente do *K-Fold* tradicional, o *Stratified K-Fold* preserva a proporção original das classes em todas as divisões, sendo particularmente eficaz na avaliação de conjuntos de dados com distribuição desbalanceada entre as classes.

Além dos resultados obtidos pelas métricas mencionadas, registra-se também o tempo de processamento de cada modelo em cada iteração, permitindo uma análise comparativa não apenas da capacidade preditiva, mas também da eficiência computacional dos algoritmos. Ressalta-se que todos os algoritmos previamente descritos são executados em cada iteração.

F. Otimização de hiperparâmetros

A otimização dos hiperparâmetros de modelos de ML é uma etapa fundamental para maximizar o desempenho e a capacidade de generalização dos algoritmos. Neste estudo, foi empregado o *Grid Search*. Essa técnica opera de forma exaustiva, explorando cada combinação predefinida de valores para os hiperparâmetros. Embora, se bem configurado e dependendo do tamanho do espaço de busca, possa garantir a identificação do ótimo, sua natureza combinatória resulta em um custo computacional que cresce exponencialmente com o aumento do número de hiperparâmetros, tornando-o inviável para modelos complexos ou espaços de busca amplos.

III. Resultados e discussão

As análises foram conduzidas de modo a avaliar o desempenho em relação às métricas e ao tempo de processamento dos algoritmos. Além disso, foram considerados três cenários distintos: o modelo completo, o modelo reduzido por PCA e o modelo reduzido utilizando FA. Os resultados encontrados serão detalhados a seguir.

A) Tratamento dos dados

Antes da aplicação dos modelos, os dados foram submetidos a um processo de tratamento que incluiu a normalização das variáveis preditoras para garantir comparabilidade entre escalas e a remoção de observações com dados faltantes, a fim de evitar distorções nas análises. Essas etapas visaram melhorar a qualidade dos dados e a performance dos algoritmos de ML.

B) Delineamento de experimentos

Com intuito de equilibrar custo computacional e qualidade dos resultados, estabeleceu-se um limite máximo de 256 iterações. Assim, para experimentos com até oito fatores, é aplicado o delineamento fatorial completo. A partir de nove fatores, recorre-se ao delineamento fatorial fracionado, uma vez que o aumento exponencial no número de combinações eleva substancialmente o tempo de processamento, muitas vezes sem ganhos proporcionais em termos de desempenho do modelo.

C) Redução de dimensionalidade

Para mostrar a importância da aplicação de técnicas de redução de dimensionalidade, também é importante considerar o custo computacional associado à geração de combinações de variáveis em planejamentos fatoriais completos, visto que esse a quantidade de iterações tende a aumentar o tempo de processamento do modelo. Por exemplo, o *dataset Chemical* apresenta originalmente 17 variáveis explicativas, o que demandaria 131.072 iterações para um planejamento fatorial completo envolvendo todas as combinações possíveis de variáveis (2^{17}). No entanto, após a aplicação de técnicas como PCA ou FA, o número de variáveis foi reduzido para apenas 7, o que resulta em 128 iterações, uma redução expressiva na carga computacional. Além disso, com essa redução, não há a necessidade de realizar fatorial fracionado, visto que a quantidade de iterações foi inferior ao definido anteriormente.

O impacto da redução de dimensionalidade também é expressivo em outros conjuntos de dados. No caso da base *Bean*, que originalmente possui 16 variáveis, seriam necessárias 65.536 iterações no planejamento fatorial completo. Com a redução para apenas 3 variáveis via PCA ou FA, esse número cai para apenas 8 iterações. Já nos *datasets Rice* e *Raisin*, ambos com 7 variáveis originais (128 iterações), a redução para 2 variáveis torna viável a realização de um planejamento com apenas 4 combinações. O *dataset ForestRain* reduziu para 5 e *Vehicle* reduziu para 9 *features.*

Esses resultados demonstram que a redução de dimensionalidade, além de contribuir para a simplificação dos modelos e a mitigação de problemas como multicolinearidade, é também uma boa estratégia para viabilizar análises mais complexas, como a busca exaustiva de melhores combinações de variáveis e otimização de hiperparâmetros. Portanto, a aplicação combinada de PCA ou FA atrelada ao planejamento de experimentos (DOE) não apenas reduz o espaço de busca, mas torna toda a estratégia de modelagem mais eficiente e robusta, permitindo inúmeras combinações de *features* já transformadas. Ao preservar a maior parte da variabilidade dos dados com um número significativamente menor de variáveis, essa abordagem permite explorar, com maior profundidade, o desempenho de múltiplas configurações de modelos, inclusive em cenários em que o uso de todas as variáveis seria computacionalmente inviável.

D) Modelos de ML

A Tabela I apresenta a melhor solução para cada uma das bases de dados analisadas, considerando três abordagens distintas: a base completa (sem redução), a redução via PCA e FA. A métrica adotada para seleção do melhor modelo foi o *F1-Score*, por representar de forma equilibrada tanto a precisão quanto o *recall*, sendo particularmente relevante em contextos com possível desequilíbrio entre as classes.

Para a base *Rice*, o melhor desempenho foi obtido com o modelo de LR, para a base original, utilizando apenas quatro *features* e alcançando um *F1-Score* de 0,9309. No caso da base Chemical, ao considerar os dados completos, observou-se a ocorrência de *overfitting* nos modelos LDA, RF e LGBM. Com a exclusão desses modelos da análise, destacaram-se como melhores alternativas o LR, *XGBoost* e MLP, todos apresentando *F1-Score* igual a 0,9889 para o modelo completo.

Para a base *Bean*, o modelo de melhor desempenho foi o RF, com sete variáveis selecionadas e um F1-Score de 0,8967. Para *ForestRain*, o melhor resultado foi obtido com o modelo LR (F1 = 0,9185) utilizando os dados sem redução de dimensionalidade. Para *Vehicle,* o melhor desempenho foi obtido com o LGBM (F1 = 0,7558) também para os dados originais. Por fim, na base *Raisin*, o melhor resultado foi alcançado com o modelo MLP aplicado sobre dados reduzidos via PCA, com apenas duas variáveis de entrada, resultando em um *F1-Score* de 0,8710.

De forma geral, observa-se que os melhores desempenhos são obtidos com o modelo original; entretanto, a diferença em relação aos resultados obtidos com a redução de dimensionalidade indica que é possível alcançar desempenhos muito semelhantes utilizando um número significativamente menor de variáveis.

TABELA I. DESEMPENHO DO MELHOR MODELO

Original						
	Rice	***Chemical***	***Bean***	***Raisin***	***ForestRain***	***Vehicle***
Iteração	46	83	116	125	24	167
Num *Features*	4	5	7	5	4	8
Features	x_2,x_3, x_4,x_6	x_2,x_3,x_4, x_7,x_{14}	x_3,x_4,x_5,x_6, x_8,x_9,x_{16}	x_3,x_4,x_5, x_6,x_7	F_2,F_7, F_{10},F_{12}	x_1,x_2,x_4,x_5, x_7,x_8,x_{12},x_{13}
A	0,9310	0,9889	0,8968	0,8589	0,9384	0,7589
P	0,9310	0,9900	0,8969	0,8618	0,9002	0,7555
R	0,9310	0,9889	0,8968	0,8589	0,9384	0,7589
F1	0,9309	0,9889	0,8967	0,8586	0,9185	0,7558
Melhor método	LR	LR, XGBoost e MLP	RF	LR	LR	LGBM
PCA						
	Rice	***Chemical***	***Bean***	***Raisin***	***ForestRain***	***Vehicle***
Iteração	4	55	8	4	11	185
Num *Features*	2	4	3	2	2	7
Features	PC_1,PC_2	PC_1,PC_3, PC_5,PC_6	PC_1,PC_2,PC_3	PC_1,PC_2	PC_1,PC_4	PC_1,PC_2,PC_3,PC_5, PC_6,PC_8,PC_9
A	0,9252	0,9765	0,8844	0,8711	0,8600	0,7293
P	0,9254	0,9812	0,8849	0,8722	0,8288	0,7255
R	0,9252	0,9765	0,8844	0,8711	0,8600	0,7293
F1	0,9252	0,9763	0,8837	0,8710	0,8422	0,7258
Melhor método	LR	XGBoost	MLP	MLP	LR	XGB
FA						
	Rice	***Chemical***	***Bean***	***Raisin***	***ForestRain***	***Vehicle***
Iteração	4	36	8	4	28	185
Num *Features*	2	3	3	2	4	7
Features	F_1,F_2	F_1,F_2,F_6	F_1,F_2,F_3	F_1,F_2	F_1,F_2,F_4,F_5	F_1,F_2,F_3,F_5, F_6,F_8,F_9
A	0,9252	0,9765	0,8844	0,8700	0,8849	0,7340
P	0,9254	0,9790	0,8849	0,8714	0,8530	0,7320
R	0,9252	0,9765	0,8844	0,8700	0,8849	0,7340
F1	0,9252	0,9764	0,8837	0,8699	0,8681	0,7311
Melhor método	LR	LR	MLP	MLP	RF	XGB

TABELA II. VALOR MÉDIO DAS MÉTRICAS EM CADA CENÁRIO

Original						
Métodos	***Rice***	***Chemical***	***Bean***	***Raisin***	***ForestRain***	***Vehicle***
KNN	0,8855	0,9623	0,7605	0,8023	0,8443	0,5688
LR	0,8456	0,9712	0,7699	0,7870	0,8861	0,5309
LDA	0,9168	0,9851	0,8931	0,8467	0,8393	0,5173
RF	0,9053	0,9898	0,9097	0,8391	0,8951	0,5961
XGBoost	0,9028	0,9800	0,9101	0,8314	0,8844	0,5922
LightGBM	0,9057	0,9835	0,9108	0,8319	0,8848	0,5950
MLP	0,5780	0,6692	0,4307	0,4857	0,6178	0,5231
PCA						
Métodos	***Rice***	***Chemical***	***Bean***	***Raisin***	***ForestRain***	***Vehicle***
KNN	0,7936	0,8163	0,5766	0,7458	0,6914	0,5317
LR	0,8071	0,7574	0,5878	0,7501	0,6844	0,4983
LDA	0,8069	0,7563	0,5840	0,7474	0,6746	0,4960
RF	0,7652	0,8117	0,5694	0,7078	0,6875	0,5406
XGBoost	0,8011	0,7998	0,6017	0,7382	0,6801	0,5309
LightGBM	0,8052	0,7970	0,6024	0,7347	0,6811	0,5322
MLP	0,8094	0,5592	0,6067	0,7422	0,5293	0,4957
FA						
Métodos	***Rice***	***Chemical***	***Bean***	***Raisin***	***ForestRain***	***Vehicle***
KNN	0,7994	0,8165	0,5766	0,7310	0,6751	0,5563
LR	0,8220	0,7875	0,5878	0,7547	0,6889	0,5201
LDA	0,8224	0,7870	0,5840	0,7556	0,6791	0,5110
RF	0,7602	0,8083	0,5694	0,7060	0,6871	0,5741
XGBoost	0,8140	0,7993	0,6017	0,7187	0,6819	0,5652
LightGBM	0,8161	0,7905	0,6024	0,7198	0,6827	0,5685
MLP	0,8215	0,5449	0,6067	0,7471	0,5011	0,5091

E) Análise de desempenho

A Tabela II apresenta os resultados médios das quatro métricas de avaliação (Acurácia, Precisão, *Recall* e *F1-Score*), obtidas a partir do planejamento fatorial, no qual múltiplas iterações foram realizadas para testar diferentes combinações de variáveis de entrada. Cada valor apresentado representa, portanto, a média do desempenho do modelo ao longo de todas essas iterações, oferecendo uma visão robusta da performance geral de cada técnica nas três diferentes configurações.

Para a base *Chemical*, os modelos com dados originais apresentaram os melhores resultados, com destaque para o *RF* (0,9898), seguido de LDA (0,9851), e LGBM (0,9835). Em contraste, ao utilizar FA e PCA os desempenhos foram consideravelmente menores com FA (0,8083) e PCA (0,8117) utilizando *Random Forest*. Isso pode indicar que a natureza da base *Chemical* demanda maior riqueza informacional nas variáveis, a qual é comprometida pelas técnicas de redução. No caso das bases *Raisin*, *ForestRain* e *Vehicle,* embora o desempenho global seja mais modesto em relação às demais bases, os melhores resultados também são obtidos com a base original.

Na base *Bean*, o método RF mais uma vez se sobressai na base original (0,9097), juntamente com *XGBoost* (0,9101) e LGBM (0,9108), todos com desempenho médio superior a 0,90. Ao aplicar FA e PCA, observa-se uma redução expressiva nos valores médios, reforçando a ideia de que os modelos se beneficiam das variáveis completas nessa base específica. Embora a aplicação de PCA e FA resulte em conjuntos de escores distintos, nesta base específica ambos os métodos produziram escores suficientemente semelhantes para que os resultados dos modelos de regressão fossem idênticos. Esse comportamento não se repetiu em outras bases analisadas, sugerindo que a estrutura de correlações da base em questão favorece convergência entre os métodos de redução de dimensionalidade.

É importante destacar que os resultados apresentados nesta seção correspondem à média dos desempenhos obtidos ao longo de todas as iterações do planejamento fatorial, nas quais diferentes combinações de variáveis foram avaliadas. Esse tipo de abordagem, embora ofereça uma visão global da estabilidade e robustez dos modelos, tende a penalizar os algoritmos nas iterações em que a quantidade ou a qualidade das variáveis selecionadas não foi suficiente para uma boa classificação. Como consequência, o desempenho médio pode ser inferior àquele obtido na melhor iteração isolada, apresentada anteriormente na Tabela I. Essa diferença evidencia o impacto da seleção adequada de variáveis sobre a performance dos modelos. Ao comparar os resultados médios com os obtidos nas melhores configurações de cada base, observa-se que a escolha estratégica da combinação de variáveis pode melhorar significativamente o desempenho final. Isso é observado, por exemplo, na base *Raisin*, em que a utilização de apenas duas variáveis principais via PCA, associadas ao modelo MLP, na Tabela I, proporcionou um *F1-Score* superior ao valor médio obtido com o mesmo método para todas iterações (Tabela II).

A análise do tempo de processamento revelou diferenças expressivas entre os modelos de classificação testados, sobretudo quando comparados os dados originais com as bases reduzidas por FA e PCA (Fig. 2). De modo geral, os modelos aplicados sobre os dados originais apresentaram os maiores tempos de execução, destacando-se os algoritmos baseados em árvores, como RF e *LightGBM*. Em particular, o RF demonstrou o maior custo computacional, com tempos superiores a 50 segundos em algumas bases, como a *Chemical*. Esse comportamento é esperado, dada a natureza do algoritmo e a alta dimensionalidade dos dados originais.

Na base *Vehicle*, os tempos de processamento medidos para os dados originais e os reduzidos por PCA e FA pareçam visualmente próximos, esse comportamento se deve principalmente à escala adotada na Fig. 2, que comprime as diferenças entre as abordagens. No entanto, os valores reais confirmam que houve reduções consistentes no tempo de processamento com a aplicação das técnicas de redução de dimensionalidade. No caso do LGBM, por exemplo, o tempo caiu de 390 segundos (dados originais) para 320 segundos com FA (uma redução de 17,9%) e para 335 segundos com PCA (uma redução de 14,1%).

Em todos os modelos e conjuntos de dados analisados, a FA contribuiu para ganhos em eficiência, permitindo que algoritmos como LR, LDA e KNN fossem executados rapidamente. Mesmo os modelos com maior demanda computacional, como RF e *LightGBM*, apresentaram diminuições nos tempos de execução, indicando que a simplificação da estrutura dos dados tem impacto positivo no desempenho computacional.

No cenário com PCA, também se verificou uma queda nos tempos de execução. Embora os ganhos tenham sido, em alguns casos, ligeiramente inferiores aos obtidos com FA, a redução ainda foi relevante. Analisando de forma global os três cenários, observa-se que as bases *Vehicle* e *Bean* apresentaram os maiores tempos de processamento. Isso ocorre devido à complexidade dos dados, sendo um volume maior de dados e/ou de classes.

Os resultados indicam que a escolha por técnicas de redução de dimensionalidade pode ser determinante na viabilidade computacional de modelos mais robustos. Nesse contexto, a FA se destacou por proporcionar, em média, os menores tempos de execução e com métricas de avaliação altas. Esses achados reforçam a importância de considerar não apenas a acurácia dos modelos, mas também seu custo

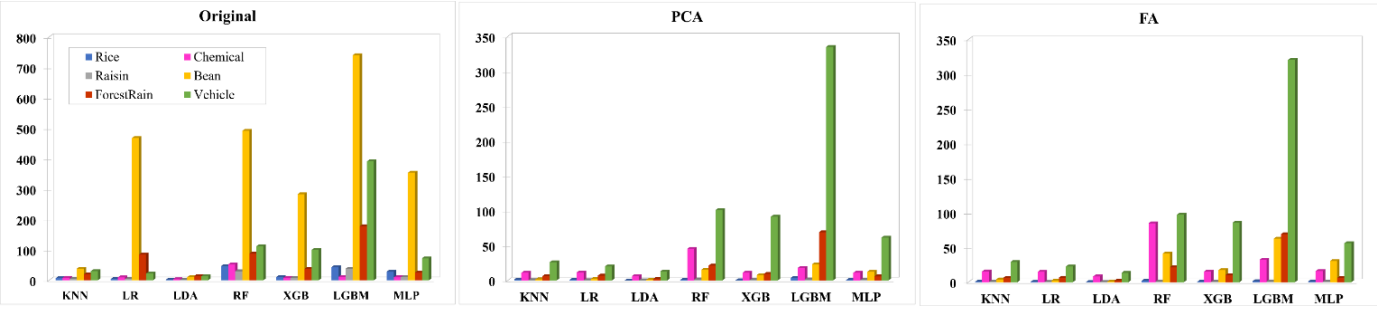

Fig. 2. Tempos de processamento dos modelos de classificação

computacional, especialmente em aplicações onde o tempo de resposta é um fator crítico.

F) Otimização de hiperparâmetros

A análise dos efeitos da otimização de hiperparâmetros via *Grid Search* demonstra que, de forma geral, os ganhos de desempenho foram discretos para a maioria dos modelos (Tabela III). Métodos lineares como LDA e LR mantiveram desempenho constante, com variações nulas ou desprezíveis nas métricas de avaliação, independentemente da base de dados ou técnica de redução de dimensionalidade utilizada. Em contrapartida, modelos mais complexos, como MLP, RF e *LightGBM*, mostraram maior sensibilidade à otimização, com destaque para o MLP na base *Rice* com dados originais, que apresentou uma melhoria média de 0,1727 nos escores. Já o *LightGBM* na base *Bean* (dados originais) alcançou incremento de 0,3496, o maior observado entre todos os cenários.

Para a base *Chemical*, observou-se que a aplicação do *GridSearch* para ajuste de hiperparâmetros não resultou em melhorias nas métricas de avaliação. Isso ocorreu porque, mesmo antes da otimização, os modelos já apresentavam desempenho elevado, o que limitou o potencial de ganho adicional. Além disso, o processo de validação cruzada reforçou a estabilidade dos resultados, evidenciando que os hiperparâmetros utilizados já se encontravam em uma região adequada do espaço de busca.

Apesar dessas melhorias pontuais, é importante destacar o custo computacional associado à etapa de otimização. Modelos como *LightGBM,* RF e MLP mantêm tempos de execução significativamente mais elevados, mesmo nas bases com menor volume de dados. Por exemplo, o tempo médio para otimização do *LightGBM* na base *Rice* (dados com PCA) foi de aproximadamente 10,43 segundos, enquanto na base *Bean* (dados originais), mesmo após o ajuste, o tempo foi superior a 52,75 segundos. Além disso, com exceção da base *Chemical,* observam-se casos de piora no desempenho após a otimização aplicados às bases reduzidas por PCA ou FA, demonstrando que nem sempre a otimização de hiperparâmetros proporciona melhoria no tempo de processamento.

Dessa forma, conclui-se que, embora o *Grid Search* possa contribuir para incrementos marginais em desempenho, sua aplicação deve ser criteriosa. O custo computacional pode não justificar os ganhos, principalmente em contextos de produção ou quando o tempo de resposta do modelo é uma variável crítica. A decisão de incluir ou não a etapa de otimização deve, portanto, ser pautada em análises conjuntas de performance e eficiência.

IV. CONCLUSÃO

Este estudo explorou o impacto da seleção de variáveis e da redução de dimensionalidade sobre o desempenho e a eficiência computacional de modelos de classificação. Foram comparadas abordagens com base original, PCA e FA em seis conjuntos de dados de âmbitos distintos, utilizando planejamento fatorial para testar múltiplas combinações de variáveis. Os resultados evidenciam que a escolha adequada das variáveis tem papel crucial na construção de modelos mais

TABELA III. VARIAÇÃO NO DESEMPENHO E TEMPO MÉDIO DE PROCESSAMENTO COM GRID SEARCH

	Rice						*Chemical*					
	Original		PCA		FA		Original		PCA		FA	
Método	**Melhoria**	**Tempo**	**Melhoria**	**Tempo**	**Melhoria**	**Tempo**	**Melhoria**	**Tempo**	**Melhoria**	**Tempo**	**Melhoria**	**Tempo**
KNN	0,0141	0,1516	-0,0032	0,3544	-0,0014	0,3499	0,0000	0,0712	0,0000	0,2342	0,0000	0,2641
LR	0,0000	0,0920	0,0000	0,1822	0,0000	0,1793	0,0108	0,0598	0,0000	0,1974	0,0000	0,2219
LDA	0,0000	0,0187	0,0000	0,0450	0,0000	0,0381	0,0000	0,0112	0,0000	0,0582	0,0000	0,0479
RF	0,0042	3,5669	0,0037	6,6059	-0,0005	5,9602	0,0000	1,2105	0,0000	4,5455	0,0000	4,7162
XGB	0,0456	0,5264	0,0013	0,9149	0,0021	0,9826	0,0000	0,1439	0,0000	0,5615	0,0000	0,6458
LGBM	0,0173	7,6114	0,0042	10,4255	0,0005	8,9027	0,0000	0,5000	0,0000	1,1898	0,0000	1,3412
MLP	0,1727	2,9438	0,0016	6,2875	0,0023	6,3722	-0,0108	0,3663	0,0000	1,2695	0,0109	1,5656
	Bean						*Raisin*					
	Original		PCA		FA		Original		PCA		FA	
Método	**Melhoria**	**Tempo**	**Melhoria**	**Tempo**	**Melhoria**	**Tempo**	**Melhoria**	**Tempo**	**Melhoria**	**Tempo**	**Melhoria**	**Tempo**
KNN	0,0116	0,0340	-0,0075	0,0354	-0,0065	0,0766	0,0221	0,1416	-0,0133	0,2277	-0,0178	0,2154
LR	0,0002	0,6927	-0,0001	0,0182	0,0003	0,0752	0,0000	0,0815	0,0047	0,1692	0,0047	0,1894
LDA	0,0000	0,0032	0,0000	0,0045	0,0000	0,0080	0,0000	0,0279	0,0000	0,0479	0,0000	0,0355
RF	0,0040	1,1929	0,0043	0,6606	0,0001	1,4523	0,0036	2,2301	0,0101	3,8429	0,0022	3,8262
XGB	0,0053	0,6135	-0,0007	0,0915	0,0013	0,8781	0,0202	0,5926	0,0111	0,6052	0,0021	0,9577
LGBM	0,3496	5,2759	0,0040	1,0426	0,0094	5,7938	0,0160	6,1575	0,0046	5,2954	0,0056	5,8944
MLP	0,0037	2,0722	-0,0006	0,6287	0,0013	2,3747	0,0268	0,8072	0,0014	2,1206	0,0088	3,0765
	ForestRain						*Vehicle*					
	Original		PCA		FA		Original		PCA		FA	
Método	**Melhoria**	**Tempo**	**Melhoria**	**Tempo**	**Melhoria**	**Tempo**	**Melhoria**	**Tempo**	**Melhoria**	**Tempo**	**Melhoria**	**Tempo**
KNN	-0,0109	2,3637	-0,0258	0,2468	-0,0120	0,4284	0,0199	0,2822	0,0000	0,2536	-0,0124	0,3150
LR	-0,0054	1,4897	0,0000	0,2689	0,0009	0,2951	0,0015	0,1667	0,0000	0,1764	0,0001	0,1902
LDA	0,0000	0,0754	0,0000	0,0482	0,0000	0,0631	0,0000	0,0595	0,0000	0,0470	0,0000	0,0336
RF	0,0000	2,4276	-0,0258	4,8832	-0,0124	5,4891	0,0000	3,4607	0,0116	3,7057	-0,0014	4,2713
XGB	0,0000	0,9004	0,0000	1,8689	0,0000	2,4063	-0,0006	4,4093	0,0037	2,9523	0,0136	3,5888
LGBM	0,0240	7,7965	0,0470	22,0237	0,0000	24,2689	0,0081	19,8537	0,0041	17,7764	-0,0002	19,9713
MLP	0,0390	1,1294	0,0172	2,4074	0,2864	2,1185	0,0633	3,1700	0,0298	3,7673	0,0262	2,7185

robustos, influenciando diretamente as métricas de acurácia, precisão, *recall* e F1-Score.

Observou-se também que a redução de dimensionalidade com PCA e FA permitiu alcançar desempenhos similares ou, em alguns casos, superiores aos obtidos com a base completa, utilizando um número significativamente menor de variáveis. Isso não apenas simplificou os modelos, mas também reduziu o espaço de busca do DOE e os custos computacionais, tornando possível explorar múltiplas configurações de forma viável. A comparação com os melhores resultados por iteração mostrou que a combinação ideal de *features* é essencial para alcançar o desempenho máximo.

A aplicação de *Grid Search* proporcionou pequenos ganhos de desempenho em alguns modelos, com custos computacionais variáveis. Modelos como RF, *LightGBM* e MLP apresentaram ganhos ligeiramente expressivos, porém com um tempo de processamento consideravelmente maior. Assim, este trabalho reforça a importância de estratégias que equilibrem desempenho e eficiência computacional, como o uso combinado de técnicas de redução de dimensionalidade, delineamento de experimentos e otimização de hiperparâmetros.

Por fim, considerando o crescente volume de dados e a demanda por soluções computacionais mais sustentáveis, futuros trabalhos podem aprofundar a investigação de métodos avançados de seleção de variáveis e redução de dimensionalidade que conciliem alto desempenho com baixo custo computacional. Abordagens como o *lightweight learning* mostram-se promissoras para reduzir custos reais em aplicações industriais e ambientais, em que a eficiência energética e a otimização de recursos vêm ganhando destaque.

AGRADECIMENTOS

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) e da Fundação de Amparo à Pesquisa do Estado de Minas Gerais (FAPEMIG).

REFERÊNCIAS

[1] E. E. M. G. Alez, J. A. R. Ortiz, and B. A. G. B. Án, “Machine Learning Models for Cancer Type Classification with Unstructured Data,” *Comput. y Sist.*, vol. 24, no. 2, pp. 403–411, 2020, doi: 10.13053/CyS-24-2-3367.

[2] A. Alsharkawi, M. Al-Fetyani, M. Dawas, H. Saadeh, and M. Alyaman, “Poverty classification using machine learning: The case of Jordan,” *Sustain.*, vol. 13, no. 3, pp. 1–16, 2021, doi: 10.3390/su13031412.

[3] I. H. Sarker, A. S. M. Kayes, and P. Watters, “Effectiveness analysis of machine learning classification models for predicting personalized context-aware smartphone usage,” *J. Big Data*, vol. 6, no. 1, 2019, doi: 10.1186/s40537-019-0219-y.

[4] G. Zhang, C. D. Carrasco, K. Winsler, B. Bahle, F. Cong, and S. J. Luck, “Assessing the effectiveness of spatial PCA on SVM-based decoding of EEG data,” *Neuroimage*, vol. 293, no. February, p. 120625, 2024, doi: 10.1016/j.neuroimage.2024.120625.

[5] A. V. S. Xavier, R. C. Almeida, L. D. Coelho, and J. F. Martins-Filho, “Classification-Model Applied to Routing Problem in Flexible-Grid Optical Networks,” *IEEE Trans. Netw. Serv. Manag.*, vol. PP, no. August 2024, p. 1, 2025, doi: 10.1109/TNSM.2025.3556770.

[6] I. Belcic, “What is classification in machine learning?” [Online]. Available: https://www.ibm.com/think/topics/classification-machine-learning

[7] G. An *et al.*, “Comparison of Machine-Learning Classification Models for Glaucoma Management,” *J. Healthc. Eng.*, vol. 2018, 2018, doi: 10.1155/2018/6874765.

[8] Y. Chen, X. Du, and M. Guo, “Self-paced ensemble for constructing an efficient robust high-performance classification model for detecting mineralization anomalies from geochemical exploration data,” *Ore Geol. Rev.*, vol. 157, no. February, p. 105418, 2023, doi: 10.1016/j.oregeorev.2023.105418.

[9] F. Wang *et al.*, “Generative adversarial networks and convolutional neural networks based weather classification model for day ahead short-term photovoltaic power forecasting,” *Energy Convers. Manag.*, vol. 181, no. December 2018, pp. 443–462, 2019, doi: 10.1016/j.enconman.2018.11.074.

[10] C. Bulut and E. Arslan, “Comparison of the impact of dimensionality reduction and data splitting on classification performance in credit risk assessment,” *Artif. Intell. Rev.*, vol. 57, no. 9, pp. 1–23, 2024, doi: 10.1007/s10462-024-10904-1.

[11] C. T. Akpinar, Ö. Koşar, and A. Durdu, “Enhancing the Performance of Machine Learning Classification Models,” *2025 24th Int. Symp. INFOTEH-JAHORINA, INFOTEH 2025 - Proc.*, no. March, pp. 19–21, 2025, doi: 10.1109/INFOTEH64129.2025.10959301.

[12] D. A. Gzar, A. M. Mahmood, and M. K. Abbas, “A Comparative Study of Regression Machine Learning Algorithms: Tradeoff Between Accuracy and Computational Complexity,” *Math. Model. Eng. Probl.*, vol. 9, no. 5, pp. 1217–1224, 2022, doi: 10.18280/mmep.090508.

[13] P. Ziolkowski, “Computational Complexity and Its Influence on Predictive Capabilities of Machine Learning Models for Concrete Mix Design,” *Materials (Basel).*, vol. 16, no. 17, 2023, doi: 10.3390/ma16175956.

[14] M. Maree and W. Shehada, “Optimizing Curriculum Vitae Concordance: A Comparative Examination of Classical Machine Learning Algorithms and Large Language Model Architectures,” *Ai*, vol. 5, no. 3, pp. 1377–1390, 2024, doi: 10.3390/ai5030066.

[15] K. Maharana, S. Mondal, and B. Nemade, “A review: Data pre-processing and data augmentation techniques,” *Glob. Transitions Proc.*, vol. 3, no. 1, pp. 91–99, 2022, doi: 10.1016/j.gltp.2022.04.020.

[16] T. Wagner, D. Guhl, and B. Langhals, “The Impact of Data Preparation and Model Complexity on the Natural Language Classification of Chinese News Headlines,” *Algorithms*, vol. 17, no. 4, 2024, doi: 10.3390/a17040132.

[17] T. Wu, Y. Xiao, M. Guo, and F. Nie, “A General Framework for Dimensionality Reduction of K-Means Clustering,” *J. Classif.*, vol. 37, no. 3, pp. 616–631, 2020, doi: 10.1007/s00357-019-09342-4.

[18] U. Buatoom and M. U. Jamil, “Improving Classification Performance with Statistically Weighted Dimensions and Dimensionality Reduction,” *Appl. Sci.*, vol. 13, no. 3, 2023, doi: 10.3390/app13032005.

[19] M. Schneider, N. Greifzu, L. Wang, C. Walther, A. Wenzel, and P. Li, “An end-to-end machine learning approach with explanation for time series with varying lengths,” *Neural Comput. Appl.*, vol. 36, no. 13, pp. 7491–7508, 2024, doi: 10.1007/s00521-024-09473-9.

[20] M. Koklu and I. Cinar, “Classification of Rice Varieties Using Artificial Intelligence Methods,” *Int. J. Intell. Syst. Appl. Eng.*, vol. 7, no. 3, pp. 188–194, 2019, doi: 10.18201/ijisae.2019355381.

[21] M. Koklu and I. A. Ozkan, “Multiclass classification of dry beans using computer vision and machine learning techniques,” *Comput. Electron. Agric.*, vol. 174, 2020, doi: 10.1016/j.compag.2020.105507.

[22] “Chemical composition of ceramic samples.” [Online]. Available: https://archive.ics.uci.edu/dataset/583/chemical+composition+of+ceramic+samples

[23] İ. CINAR, M. KOKLU, and Ş. TAŞDEMİR, “Kuru Üzüm Tanelerinin Makine Görüşü ve Yapay Zeka Yöntemleri Kullanılarak Sınıflandırılması,” *Gazi J. Eng. Sci.*, vol. 6, no. 3, pp. 200–209, 2020, doi: 10.30855/gmbd.2020.03.03.

[24] F. Abid and N. Izeboudjen, “Predicting Forest Fire in Algeria Using Data Mining Techniques: Case Study of the Decision Tree Algorithm,” *Adv. Intell. Syst. Comput.*, vol. 1105 AISC, pp. 363–370, 2020, doi: 10.1007/978-3-030-36674-2_37.

[25] J. P. Siebert, “Vehicle Recognition Using Rule Based Methods.,” 1987. [Online]. Available: https://archive.ics.uci.edu/dataset/149/statlog+vehicle+silhouettes