

Aprendizado de Máquina para a Otimização da Obtenção de Resultados em Simulações de Defesa Aeroespacial

Lucas Silva Lima¹, Rafael Hoffmann Giannico¹, Denys Derlian Carvalho Brito¹,
Antonio Gustavo Silveira Dantas¹, João Paulo De Andrade Dantas¹
¹Instituto Tecnológico de Aeronáutica, São José dos Campos/SP

Resumo— Este estudo investiga a implementação de técnicas de *Machine Learning* (Aprendizado de Máquina, ML) para acelerar a aquisição dos resultados em simulações aeroespaciais executadas pelo *software* Ambiente de Simulação Aeroespacial (ASA). Nesse contexto, simulações que modelam, em alguma medida, comportamentos estocásticos naturalmente requerem múltiplas iterações para atingir a convergência estatística, prolongando o tempo de processamento. Portanto, é proposta uma abordagem baseada em ML, utilizando técnicas tanto supervisionadas quanto não supervisionadas, para construir modelos preditivos a partir de dados provenientes de simulações de cenários aeroespaciais no contexto de defesa, de modo que, uma vez treinados, estes modelos generalizem a relação entre os dados de entrada e a variável-alvo, possibilitando a inferência de resultados sem a necessidade de recorrer a extensas e demoradas novas simulações para um mesmo cenário. Este procedimento é implementado utilizando a biblioteca *AsaPy*, desenvolvida na linguagem de programação *Python*, e integrado ao serviço ASA. Os resultados evidenciam a aplicabilidade e a usabilidade do ML em contextos de simulação aeroespacial de defesa.

Palavras-Chave— Aprendizado de Máquina, Simulação, Ambiente de Simulação Aeroespacial (ASA), Análise de Dados, Apoio à Decisão.

I. INTRODUÇÃO

A simulação aeroespacial tem sido amplamente utilizada no campo da aviação e defesa há décadas. Desde os primeiros simuladores de voo desenvolvidos desde o final da Segunda Guerra Mundial (Bruzzone e Massei 2017) até os sistemas de simulação de última geração utilizados atualmente, a simulação desempenha um papel fundamental no treinamento de pilotos, no projeto de aeronaves e no planejamento de missões, oferecendo alta flexibilidade e capacidade de integração para *design* de produtos, desenvolvimento e eficiência de fabricação na indústria aeroespacial (Murphy e Perera 2002).

No entanto, a aplicação de técnicas de *Machine Learning* (Aprendizado de Máquina, ML) em simulações aeroespaciais é um desenvolvimento mais recente. Com o avanço da tecnologia de computação e o aumento da disponibilidade de dados, tornou-se possível utilizar algoritmos de ML para acelerar a obtenção de resultados em simulações complexas, resolvendo problemas complexos de otimização e aprimorando aplicações críticas de segurança (Brunton et al. 2020).

Nesse contexto, surge o Ambiente de Simulação Aeroespacial (ASA), uma plataforma virtual onde é possível

modelar e analisar o desempenho de sistemas aeroespaciais, como aeronaves, satélites e veículos espaciais (Dantas et al. 2022; Dantas, Geraldo et al. 2023). Este *software* possibilita a realização de estudos de viabilidade, testes de sistemas, análise de missão e avaliação de desempenho de projetos aeroespaciais. Ele permite que engenheiros e pesquisadores explorem diferentes configurações e trajetórias, fornecendo *insights* para o desenvolvimento e aprimoramento de tecnologias aeroespaciais (Gobi et al. 2023).

No entanto, é importante ressaltar que a aplicação de técnicas de ML em simulações aeroespaciais continua em estágio inicial. Existem desafios técnicos e práticos a serem superados, como a disponibilidade de dados de treinamento adequados e a validação dos modelos de ML em cenários realistas. Apesar desses desafios, o uso de ML em simulações aeroespaciais tem um potencial significativo para melhorar a eficiência e a precisão das simulações, contribuindo para avanços na aviação e defesa (Brunton et al. 2020).

Isto posto, em termos de planejamento e execução de missões aéreas, a simulação aeroespacial desempenha um papel importante, sobretudo, em missões críticas, nas quais precisão e eficiência são vitais. No entanto, simulações que modelam comportamentos estocásticos ou complexos podem ser extremamente demoradas, exigindo múltiplas iterações para alcançar a convergência estatística e resultados confiáveis (Davis et al. 2018). Nesse cenário, a aplicação de técnicas de ML surge como uma abordagem promissora para acelerar a obtenção de resultados em simulações aeroespaciais, permitindo uma tomada de decisão mais rápida e informada (Ni 2022).

Nesse contexto, este estudo propõe a implementação e análise do uso de ML para otimizar o processo de obtenção de dados de simulação, visando reduzir o tempo necessário para alcançar resultados eficientes outrora obtidos por métodos exigentes de tempo e de poder computacional. Dessa forma, integrando modelos de ML com dados históricos de simulações, é proposta uma metodologia de implementação de modelos de Inteligência Artificial (IA) capazes de prever resultados de simulações sem a necessidade de recorrer a novas simulações do mesmo cenário, com vistas a minimizar o custo computacional e o tempo de processamento.

O restante deste artigo está organizado da seguinte forma: Na Seção II, é feita uma revisão concisa de trabalhos anteriores que apresentam propostas relacionadas ao tema abordado, ainda que de outros domínios de

conhecimento. Logo após, na Seção III, apresenta-se a metodologia, detalhando cada etapa do processo, assim como as ferramentas e técnicas utilizadas para sua implementação. Em seguida, na Seção IV, são apresentados os resultados obtidos por meio da aplicação da metodologia. Na Seção V, discutem-se as implicações dos resultados obtidos anteriormente. Por fim, na Seção VI, apresenta-se a conclusão, além de destacarem-se sugestões para possíveis direções em pesquisas futuras nesta área.

II. TRABALHOS CORRELACIONADOS

Dentro do vasto campo da simulação e modelagem, a integração de técnicas de ML tem emergido como uma abordagem promissora para melhorar não somente a eficiência e a precisão de simulações no segmento de defesa aeroespacial, como também em áreas diversas, que incluem o planejamento urbano e análises biomédicas. A capacidade do ML de inferir e prever resultados a partir de dados provenientes de simulação oferece uma alternativa aprimorada ao uso somente de tradicionais e custosas simulações computacionais, abrindo novos caminhos para pesquisas e desenvolvimentos.

Um dos estudos revisados (Paula Martins 2022) explora a integração de técnicas de ML em sistemas de simulação baseada em agentes, com um foco particular na otimização da eficiência operacional em modelos de simulação. Este trabalho ilustra não apenas a aplicabilidade do ML em aprimorar simulações, mas também destaca sua capacidade de fornecer *insights* valiosos para a otimização de processos complexos.

Por outro lado, Filho 2021 propõe uma metodologia inovadora que utiliza ML para simplificar análises complexas de conforto térmico em escala urbana, demonstrando como o ML pode ser empregado para extrair informações relevantes de conjuntos de dados. Este estudo ressalta a versatilidade do ML, mostrando seu potencial em aplicações práticas no planejamento urbano, ao mesmo tempo que sugere uma nova direção para o uso de dados de simulação existentes.

Na esfera da medicina e biologia computacional, a aplicação de ML pode revolucionar a precisão diagnóstica e a eficiência terapêutica (Rocha 2012; Miletto 2018). Esses trabalhos não somente evidenciam o potencial do ML em desvendar complexas relações biológicas e padrões ocultos em dados de simulação, mas também mostram a rapidez com que o ML permite a implementação de novas hipóteses e terapias. A integração de ML em simulações biomédicas abre caminhos para diagnósticos mais rápidos e tratamentos personalizados, marcando um avanço significativo na otimização dos cuidados de saúde.

No contexto específico da defesa aeroespacial, a aplicação de modelos de ML para generalizar dados de simulações anteriores representa uma inovação significativa, conforme explorado neste artigo. Essa abordagem destaca-se pelo potencial de economizar recursos significativos e reduzir o tempo necessário para a análise de novos cenários de simulação.

III. METODOLOGIA

A metodologia adotada engloba as seguintes etapas: definição de caso de uso, coleta de dados, preparação

de dados, desenvolvimento e treinamento dos modelos de ML, além da avaliação. Esta abordagem estrutura o processo de desenvolvimento, implementação, treinamento e avaliação dos modelos de ML de forma a alcançar o objetivo proposto.

A. Definição do Caso de Uso

Dada a complexidade das missões aéreas e a necessidade de planejamento detalhado, o consumo de combustível figura dentre os elementos importantes para a eficiência e a segurança operacional (Milharadas 2013). Nesse escopo, o caso de uso selecionado para demonstrar a aplicação de técnicas de ML neste estudo é a previsão do consumo de combustível durante uma missão de navegação de uma aeronave de caça (Fig. 1). Embora os parâmetros de consumo de combustível sejam geralmente determinísticos e fornecidos pelos fabricantes nas especificações técnicas das aeronaves, há uma complexidade intrínseca na determinação do consumo de combustível de forma precisa e adaptada às especificidades de um cenário particular, fazendo-se necessária a simulação para a efetiva previsão e, por conseguinte, otimização.

É importante ressaltar que, devido à natureza reservada das informações relacionadas a modelos realísticos e cenários operacionais complexos, que expõem a doutrina operacional vigente, o caso de uso será restrito a um cenário simplificado. Contudo, a abordagem metodológica e a sequência de procedimentos selecionados para esta análise foram estruturadas para serem agnósticas ao grau de complexidade do cenário sob investigação. Isso permite que o método proposto seja replicável em um espectro mais amplo de situações, abrangendo desde contextos simplificados até aqueles de maior complexidade.

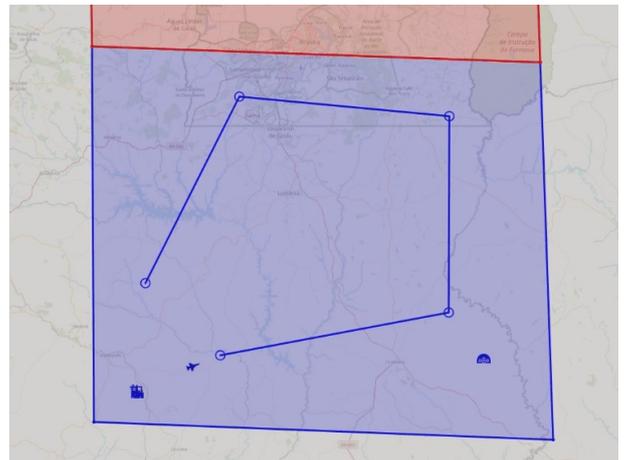


Fig. 1: Interface gráfica do ASA destinado à construção de cenários.

B. Coleta de Dados

No que tange à coleta de dados (Tabela 1), foram consideradas 4000 iterações de simulação realizadas no ASA. O cenário específico utilizado para a captação de dados descreve um voo de navegação realizado por uma aeronave de caça, com foco na execução de manobras em diversas altitudes e velocidades. De modo geral, o perfil

de voo das simulações abrange uma variação de altitude entre *Flight Level* (FL) (Agência Nacional de Aviação Civil 2024) 100 e 350, além de uma variação de velocidade entre 350 e 550 knots.

Para garantir que as amostras fossem representativas de todo o intervalo de valores possíveis para cada variável, a metodologia empregada na amostragem foi o *Latin Hypercube Sampling* (Amostragem Hiperbólica Latina, LHS), especialmente útil em simulações de Monte Carlo (McKay, Beckman e Conover 1979). O LHS é um método que busca uma melhor cobertura do espaço de busca, visto que uma abordagem puramente aleatória pode concentrar as amostras em certos intervalos. Sua principal ideia é dividir o espaço multidimensional de modo que as amostras aleatórias sejam retiradas dessas subdivisões, em vez de do espaço de busca inteiro (Husslage et al. 2011). Devido a esses benefícios, em comparação com métodos puramente aleatórios, tem sido utilizado para o delineamento de experimentos aeroespaciais e de defesa (Dantas et al. 2021a, 2021b).

TABELA 1: AMOSTRA DOS DADOS OBTIDOS DA SIMULAÇÃO

Exp.	Vel. (knot)	Alt. (ft)	Comb. consumido (lb)
244	505.22	23646.51	1065.37
411	546.05	30595.64	945.54
1480	350.66	19152.46	1115.60
2015	352.43	19110.39	1103.92
3572	465.19	12089.28	1431.66

C. Preparação dos Dados

O processo de preparação dos dados foi realizado com o auxílio da biblioteca *AsaPy* (Dantas, Silva et al. 2023). Nesta etapa, os seguintes procedimentos foram realizados:

Limpeza de Dados: Inicialmente, os dados coletados foram submetidos a uma verificação de integridade para assegurar sua qualidade e adequação. Esse processo incluiu a eliminação de registros que apresentassem variáveis com valores ausentes ou inconsistentes.

Padronização dos Valores Numéricos: Para assegurar que todas as variáveis tenham igual influência sobre os modelos, independentemente de suas escalas originais, e permitir que os dados se ajustem melhor às premissas dos algoritmos de ML, foi aplicada uma transformação de padronização, *Z-Score* (Bruce e Bruce 2019), conforme descrito em (1). Os gráficos de dispersão *Boxplot* antes e depois da padronização podem ser vistos nas Fig. 2 e 3, respectivamente.

Z-Score:

$$z(x_i) = \frac{x_i - \mu}{\sigma} \quad (1)$$

onde:

- x_i é o valor da i -ésima amostra.
- μ é a média das amostras.
- σ é o desvio padrão das amostras.

Separação dos Dados: Por fim, os dados foram divididos em conjuntos distintos para treinamento (70%), validação (15%) e teste (15%). Esta segmentação é importante para uma avaliação não enviesada da *performance* dos modelos, permitindo que o treinamento ocorra com

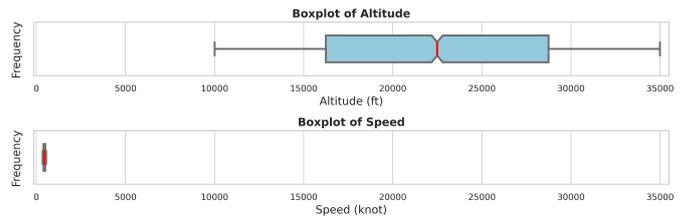


Fig. 2: *Boxplot* dos dados antes de serem padronizados.

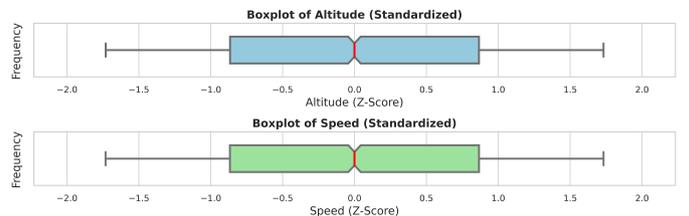


Fig. 3: *Boxplot* dos dados após serem padronizados.

uma fração do conjunto de dados, enquanto a validação e os testes são realizados em outras frações independentes.

D. Desenvolvimento dos Modelos de ML

Serão desenvolvidos três conjuntos de modelos distintos empregando técnicas de ML, cada um utilizando as variáveis preditoras obtidas da simulação (velocidade e altitude). Cada modelo será adaptado ao cenário específico em estudo, considerando as manobras planejadas e a geometria da rota.

Modelos de Regressão: Esses modelos utilizarão técnicas de aprendizado supervisionado para estimar o consumo de combustível. Os algoritmos serão Redes Neurais Artificiais (RNA) (Goodfellow, Bengio e Courville 2016) e *Extreme Gradient Boosting* (*XG Boost*) (Chen e Guestrin 2016).

Modelo de Agrupamento: Esse modelo utilizará técnicas de aprendizado não supervisionado para segmentar o regime da missão em quatro classes de *performance*, permitindo uma análise mais detalhada da operação. O algoritmo utilizado será o *K-Means* (Coates e Ng 2012).

Modelos de Classificação: Para fins de exemplificação, implementamos modelos de classificação, partindo do pressuposto que as classes identificadas no modelo de agrupamento refletem dados factuais e conhecidos. Esta abordagem tem como finalidade demonstrar explicitamente como a classificação pode reidentificar as classes associadas durante a segmentação, servindo como modelo para situações aplicáveis em casos reais. Para essa ilustração, optamos por utilizar os algoritmos RNA e *XGBoost*. Destacamos que a intenção é puramente didática, enfatizando a capacidade dos modelos de classificação de reconhecer e inferir adequadamente as categorias determinadas na etapa de agrupamento, ilustrando assim o potencial prático desses métodos de classificação em contextos analíticos reais.

E. Treinamento dos modelos de ML

Ao longo do treinamento, cada modelo desenvolvido será submetido a um processo de validação, utilizando

métricas de desempenho adequadas ao seu tipo. A saber: Raiz do Erro Quadrático Médio (*Root Mean Squared Error*, RMSE), conforme (2), para modelos de regressão; Entropia Cruzada Categórica (*Categorical Crossentropy*) para modelos de classificação, conforme (3); e Índice de Silhueta (*Silhouette Score*), conforme (4), para modelos de agrupamento.

Raiz do Erro Quadrático Médio (RMSE):

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (2)$$

onde:

- n é o número total de observações.
- y_i é o valor real da i -ésima observação.
- \hat{y}_i é o valor previsto para a i -ésima observação.

Entropia Cruzada Categórica:

$$Categorical\ Crossentropy = - \sum_{i=1}^M y_i \log(\hat{y}_i) \quad (3)$$

onde:

- M é o número de classes.
- y_i é um vetor binário indicando a classe real da i -ésima observação.
- \hat{y}_i é o vetor de probabilidades previstas para a i -ésima observação, correspondendo a cada classe.

Índice de Silhueta:

$$Silhouette\ Score = \frac{b - a}{\max(a, b)} \quad (4)$$

onde:

- a é a distância média entre uma amostra e todos os outros pontos na mesma classe.
- b é a distância média entre uma amostra e todos os outros pontos na classe mais próxima.

A validação, realizada durante o processo de treinamento de um modelo ML, é uma etapa fundamental para garantir que o modelo retenha a capacidade de realizar inferências sobre conjuntos de dados aos quais não foi exposto durante o seu aprendizado. A validação é necessária porque, nesta fase, o modelo é condicionado exclusivamente com base nos dados de treino, sem acesso aos demais de dados. O desafio é direcionar o modelo para que ele generalize as relações intrínsecas entre as variáveis de entrada e saída, em vez de apenas memorizar as peculiaridades específicas do conjunto de treino fornecido.

O sobreajuste (*overfitting*) ocorre quando um modelo aprende padrões específicos dos dados de treinamento, incluindo o ruído oriundo da variância dos dados, a tal ponto que comprometa a sua capacidade de efetuar previsões assertivas em novos conjuntos de dados (LeCun, Bengio e Hinton 2015). Para prevenir essa condição, emprega-se a técnica de parada antecipada (*Early Stopping*), que consiste em interromper o treinamento quando não há mais melhoria na métrica-alvo tomada no conjunto de

dados de validação, após um número predefinido de iterações, denominado “paciência”. Isso assegura que o modelo mantenha a capacidade de fazer previsões generalizáveis, em vez de simplesmente reproduzir os padrões aprendidos dos dados de treinamento.

O viés representa a discrepância entre as previsões de um modelo e os valores reais que não pode ser reduzida pela simples adição de mais dados, devido a simplificações excessivas nas hipóteses do modelo que limitam sua capacidade de representar a complexidade inerente ao conjunto de treinamento. Em contrapartida, a variância caracteriza-se pelo erro decorrente da sensibilidade do modelo às flutuações específicas do conjunto de treinamento, manifestando-se em um desempenho inconsistente em diferentes conjuntos de dados devido à complexidade excessiva do modelo. A otimização de um modelo de aprendizado de máquina envolve a busca por um equilíbrio entre viés e variância, visando alcançar uma *performance* consistente tanto em dados de treinamento quanto em dados de teste, que não foram utilizados durante o processo de treinamento (Murphy 2012).

A Fig. 4 apresenta um exemplo didático da evolução de um treinamento de modelo de ML, destacando sua *performance* em diferentes conjuntos de dados. Observa-se que a curva de treinamento evidencia uma diminuição contínua do erro, refletindo a capacidade do modelo de assimilar os padrões dos dados de treinamento. Em contrapartida, a curva de validação revela a eficácia do modelo ao lidar com dados que não participaram do seu processo de ajuste. Já a linha de teste, representa efetivamente o desempenho alcançado pelo modelo no conjunto de teste após a conclusão do treinamento. A tendência da curva de validação de aproximar-se assintoticamente da linha de teste indica que o modelo logrou um estado de generalização ótimo, alcançando um equilíbrio preciso entre o viés e a variância (Hastie, Tibshirani e Friedman 2009).

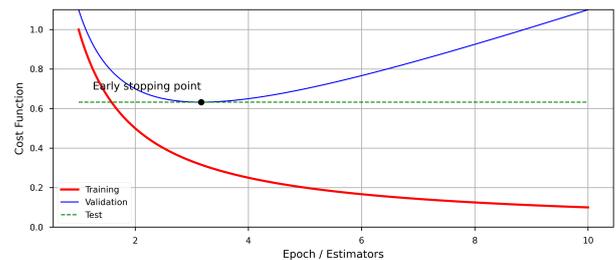


Fig. 4: Gráfico ilustrativo de *Bias-Variância*.

F. Avaliação

A eficácia dos modelos será medida por sua capacidade de prever o consumo de combustível e de classificar as missões em diferentes faixas de desempenho, utilizando este conjunto de dados de teste. Nos modelos de regressão, as métricas a serem empregadas serão a RMSE e o R^2 (Coeficiente de Determinação), este último conforme descrito em (5). Nos modelos de classificação, a métrica utilizada será a Entropia Cruzada Categórica. Por fim, no modelo de agrupamento, será aplicado o Índice de Silhueta.

Coefficiente de Determinação:

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (5)$$

onde:

- n é o número total de observações.
- y_i é o valor real da i -ésima observação.
- \hat{y}_i é o valor previsto pelo modelo para a i -ésima observação.
- \bar{y} é a média dos valores reais das observações.

G. Apresentação dos resultados

Os resultados serão apresentados de forma estruturada para cada tipo de modelo de ML utilizado, acompanhado da sua métrica de avaliação tomadas nos dados destinados ao teste. Para modelos de regressão, incluiremos amostras de dados com as respectivas previsões e valores reais de combustível consumido. Nos modelos de classificação, utilizaremos matrizes de confusão (Pedregosa et al. 2011) para destacar a precisão e acurácia das previsões. Para o agrupamento, será exibido o Gráfico de Incorporação Estocástica de Vizinhos com Distribuição t (*t-Distributed Stochastic Neighbor Embedding*, t-SNE) (Bunte et al. 2012) que proporciona uma visualização clara da distribuição dos agrupamentos e da eficácia do modelo em identificar padrões distintos nos dados. Além disso, uma tabela condensará o tempo gasto em cada etapa de modelagem e na simulação do cenário no ASA.

IV. RESULTADOS

Os resultados deste estudo foram obtidos a partir da implementação e avaliação dos três conjuntos de modelos distintos de ML propostos na Seção III.

A. Performance dos Modelos

Modelos de Regressão: Apresentaram um desempenho significativo na previsão do consumo de combustível, obtendo o melhor modelo (*XG Boost*) a RMSE desvio padrão igual a 0.03385 e R^2 igual a 0.999. A evolução da RMSE para cada fração dos dados (treinamento, validação e teste) durante o treinamento é apresentado nas Fig. 5 e 6. A Tabela 2 contém o R^2 obtido por cada modelo. A Tabela 3 apresenta a inferência do melhor modelo de regressão para os exemplos apresentados previamente na Tabela 1.

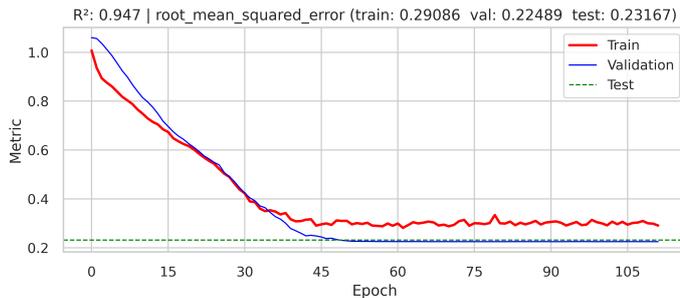


Fig. 5: Gráfico de *Bias-Variância* do modelo de regressão baseado no algoritmo RNA.

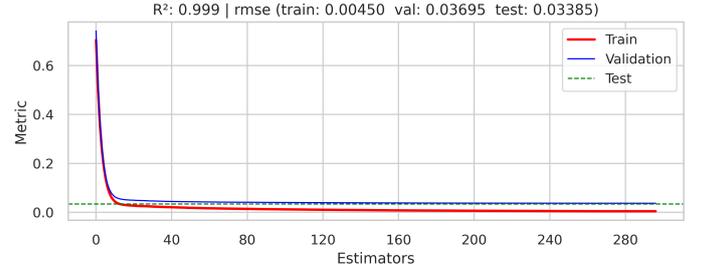


Fig. 6: Gráfico de *Bias-Variância* do modelo de regressão baseado no algoritmo *XG Boost*.

TABELA 2: DESEMPENHO DOS MODELOS DE REGRESSÃO

Algoritmo	RMSE	R^2
XG Boost	0.03385	0.999
RNA	0.23167	0.947

TABELA 3: PREVISÃO DE CONSUMO DE COMBUSTÍVEL NAS SIMULAÇÕES PREVIAMENTE AMOSTRADAS (*XG Boost*)

Exp.	Comb. cons. Real (lb)	Comb. cons. Previsto (lb)
244	1065.37	1075.48
411	945.54	944.78
1480	1115.60	1113.65
2015	1103.92	1095.03
3572	1431.66	1437.84

Modelo de Agrupamento: Segmentou efetivamente os regimes de missão em quatro classes de *performance*, obtendo um Índice de Silhueta de 0.39453. A Tabela 4 destaca essa métrica atingida e a Fig. 7 apresenta uma visualização espacial segmentada dos registros de simulação segundo a classe a qual foram associados.

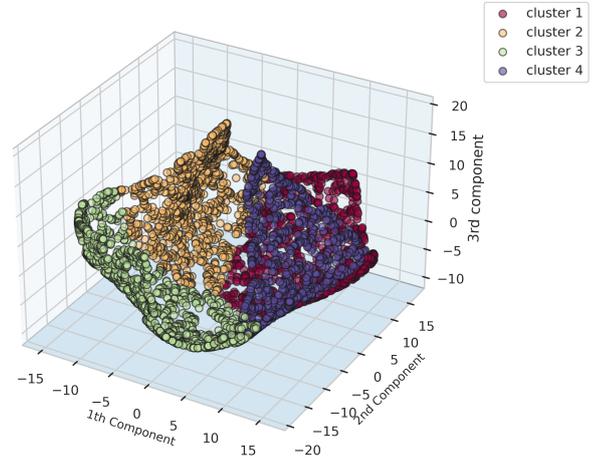


Fig. 7: Gráfico de Incorporação Estocástica de Vizinhos com Distribuição t .

TABELA 4: DESEMPENHO DO MODELO DE AGRUPAMENTO

Algoritmo	Índice de Silhueta
K-Means	0.39453

Modelos de Classificação: Conforme detalhado pelas matrizes de confusão (Fig. 9 e 11), os modelos identificaram corretamente as classes artificialmente predefinidas, obtendo o melhor modelo (*XG Boost*) a Entropia Cruzada Categórica igual a 0.02985. A evolução dessa métrica para

cada fração dos dados (treinamento, validação e teste) durante o treinamento é apresentado nas Fig. 8 e 10. A Tabela 5 destaca a Entropia Cruzada Categórica obtido por cada modelo.

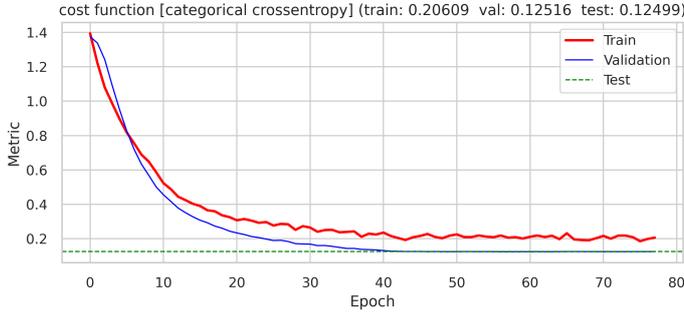


Fig. 8: Gráfico de *Bias-Variância* do modelo de classificação baseado no algoritmo RNA.

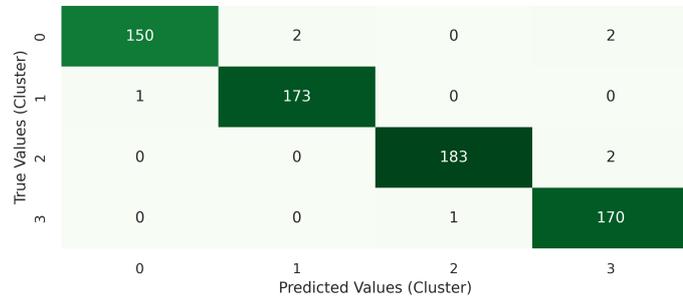


Fig. 9: Matriz de confusão do modelo de classificação baseado no algoritmo RNA.

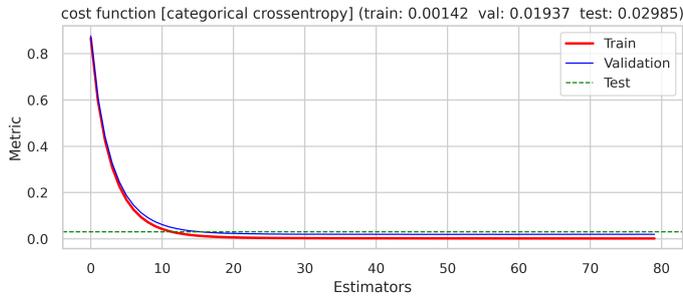


Fig. 10: Gráfico de *Bias-Variância* do modelo de classificação baseado no algoritmo *XG Boost*.

V. DISCUSSÃO

Com base nos dados das Tabelas 2 e 6, a avaliação da eficiência temporal, conforme descrito em (6), demonstrou que, ao custo dos baixos erros de inferência, a implementação dos modelos de ML para regressão resultou em uma redução significativa no tempo necessário para adquirir resultados de simulação. Especificamente, o modelo de melhor desempenho, o *XG Boost*, conseguiu diminuir o tempo de obtenção dos resultados em 99.5% (7), enquanto o segundo modelo em termos de *performance*, a RNA, registrou uma redução de 97.5% (8). Ambas as melhorias são avaliadas em relação ao método de simulação tradicional.

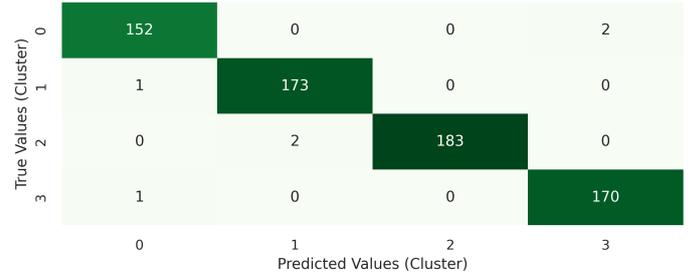


Fig. 11: Matriz de confusão do modelo de classificação baseado no algoritmo *XG Boost*.

TABELA 5: DESEMPENHO DOS MODELOS DE CLASSIFICAÇÃO

Algoritmo	Entropia Cruzada Categórica
<i>XG Boost</i>	0.02985
RNA	0.12499

TABELA 6: TEMPO DE TREINAMENTO/SIMULAÇÃO E INFERÊNCIA DOS MÉTODOS DISCUTIDOS

Método	T. de treino (s)	T. de Inferência (s)
Simulação (ASA)	48013.000	-
RNA (Reg.)	1189.263	0.009
<i>XG Boost</i> (Reg.)	225.531	0.009
K-Means (cluster.)	38.348	-
RNA (Class.)	1079.477	0.010
<i>XG Boost</i> (Class.)	200.894	0.004

Obs.: O processamento foi realizado em um computador com a especificação: SO Linux Ubuntu SMP x86_64, CPU Xeon® Gold6230R 2.10 GHz 20 Cores, RAM 62 GB.

Redução Percentual:

$$RP = 1 - \frac{T_{ML}}{T_{ASA}} \quad (6)$$

onde:

- RP é a redução percentual.
- T_{ML} é o tempo total destinado a treinamento e inferência do modelo de ML.
- T_{ASA} é o tempo total destinado a simulação do cenário no ASA.

$$RP_{XG\ Boost} = 1 - \frac{225.531 + 0.009}{48013} = 0.995 \quad (7)$$

$$RP_{RNA} = 1 - \frac{1189.263 + 0.009}{48013} = 0.975 \quad (8)$$

Retornando ao objetivo principal deste estudo, os resultados apontam uma aplicabilidade de técnicas de ML para acelerar o tempo de obtenção de resultados das simulações aeroespaciais. A redução no tempo de simulação, sem comprometer a fidelidade dos resultados, destaca o potencial das abordagens de ML para tornar mais ágil e preciso o planejamento e a execução de missões aéreas.

Uma ressalva importante na aplicação de modelos de ML em qualquer domínio, incluindo simulações aeroespaciais, é a sua tendência a se tornarem especialistas em prever cenários estritamente dentro do espectro dos dados usados durante o treinamento. Isso significa que, embora esses modelos possam ser precisos ao inferir resultados em situações que, apesar de inéditas, compartilham a mesma distribuição estatística dos dados de treinamento, sua eficácia pode ser reduzida em cenários que desviam dessa

distribuição. Essa característica sublinha a importância de uma seleção abrangente e representativa de dados de treinamento, que deve englobar a maior variedade possível do conjunto de estados do cenário em análise (James et al. 2013).

VI. CONCLUSÃO

Este estudo demonstrou que o uso de técnicas de ML pode acelerar e aprimorar o processo de obtenção de resultados de simulação aeroespacial. A implementação desses modelos não apenas reduz o tempo de planejamento e aumenta a segurança operacional, mas também abre novos caminhos para a análise e otimização de missões aéreas.

Pesquisas futuras têm o potencial de ampliar a aplicação desses modelos para cenários de combate, onde se observa o uso simultâneo de diferentes tipos de vetores, como aeronaves e sistemas de defesa antiaérea, e diversas condições, incluindo combate assimétrico e variações de tempo de resposta, com o objetivo de desenvolver e validar novas táticas operacionais. Uma segunda vertente promissora é o uso de modelos preditores para mapear a evolução do estado de um cenário, o que permitiria acelerar o processo iterativo de construção, treinamento e validação de modelos baseados em *Reinforcement Learning* (Aprendizado por Reforço, RL) (Sutton e Barto 2018). Este enfoque, especificamente, visa aprimorar um modelo que seja capaz de tomar decisões diante de um cenário específico, especialmente útil em contextos de alto dinamismo, onde o ambiente é extremamente reativo às ações dos agentes. Diferentemente do treinamento convencional, que se baseia em dados estáticos, este modelo se beneficiaria da evolução do ambiente em resposta às suas ações, guiado por uma função de recompensa. Incorporando esta capacidade, a abordagem abre caminho para o desenvolvimento de sistemas mais adaptativos e eficientes, capazes de atuar em ambientes complexos e de dinâmica parcialmente desconhecida, com notável grau de otimalidade, melhorando significativamente o processo de apoio de tomada de decisões sob condições críticas.

VII. CONTRIBUIÇÕES DOS AUTORES

Lucas S. Lima: Concepção e desenho da pesquisa; Elaboração do manuscrito.

Rafael H. Giannico: Aquisição de dados; Análise e interpretação de dados.

Denys D. C. Brito: Revisão de literatura.

Antonio G. S. Dantas: Revisão intelectual do manuscrito.

João P. A. Dantas: Aprovação final da versão submetida ao congresso.

REFERÊNCIAS

- Agência Nacional de Aviação Civil. 2024. *RBAC 01: Regulamentos Brasileiros de Aviação Civil. Definições, regras de redação e unidades de medida*. RBAC N^o 01, Emenda N^o 16. Brasília: Agência Nacional de Aviação Civil, fevereiro. Acesso em 28 de março de 2024. <https://www.anac.gov.br/assuntos/legislacao/legislacao-1/rbha-e-rbac/rbac/rbac-01>.
- Bruce, A. e P. Bruce. 2019. *Estatística prática para cientistas de dados: 50 conceitos essenciais*. Alta Books. ISBN: 9788550806037. <https://books.google.co.uk/books?id=PyrgzQEACAAJ>.
- Brunton, S., J. Kutz, Krithika Manohar, A. Aravkin, K. Morgansen, Jennifer Klemisch, Nicholas Goebel et al. 2020. Data-Driven Aerospace Engineering: Reframing the Industry with Machine Learning. *ArXiv abs/2008.10740*. <https://doi.org/10.2514/1.J060131>.
- Bruzzone, A. e M. Massei. 2017. Simulation-Based Military Training, 315–361. https://doi.org/10.1007/978-3-319-61264-5_14.
- Bunte, K., S. Haase, Michael Biehl e T. Villmann. 2012. Stochastic neighbor embedding (SNE) for dimension reduction and visualization using arbitrary divergences. *Neurocomputing* 90:23–45. <https://doi.org/10.1016/j.neucom.2012.02.034>.
- Chen, Tianqi e Carlos Guestrin. 2016. XGBoost: A Scalable Tree Boosting System. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, <https://doi.org/10.1145/2939672.2939785>.
- Coates, Adam e A. Ng. 2012. Learning Feature Representations with K-Means, 561–580. <https://doi.org/10.1007/978.3.642.35289.8.30>.
- Dantas, Joao P. A., Andre N. Costa, Diego Geraldo, Marcos R. O. A. Maximo e Takashi Yoneyama. 2021a. Engagement Decision Support for Beyond Visual Range Air Combat. Em *Proceedings of the 2021 Latin American Robotics Symposium, 2021 Brazilian Symposium on Robotics, and 2021 Workshop on Robotics in Education*, 96–101.
- . 2021b. Weapon Engagement Zone Maximum Launch Range Estimation Using a Deep Neural Network. Em *Intelligent Systems*, editado por André Britto e Karina Valdivia Delgado, 193–207. Cham: Springer. ISBN: 978-3-030-91699-2.
- Dantas, Joao P. A., Andre N. Costa, Vitor C. F. Gomes, Andre R. Kuroswiski, Felipe L. L. Medeiros e Diego Geraldo. 2022. *ASA: A Simulation Environment for Evaluating Military Operational Scenarios*. arXiv: 2207.12084 [cs.DC].
- Dantas, Joao P. A., Samara R. Silva, Vitor C. F. Gomes, Andre N. Costa, Adrisson R. Samersla, Diego Geraldo, Marcos R. O. A. Maximo e Takashi Yoneyama. 2023. *AsaPy: A Python Library for Aerospace Simulation Analysis*. arXiv: 2310.00001 [cs.MS].
- Davis, Damek, Dmitriy Drusvyatskiy, Sham Kakade e Jason D. Lee. 2018. *Stochastic subgradient method converges on tame functions*. arXiv: 1804.07795 [math.OA].

- Dantas, Joao P. A., Diego Geraldo, Andre N. Costa, Marcos R. O. A. Máximo e Takashi Yoneyama. 2023. ASA-SimaaS: Advancing Digital Transformation through Simulation Services in the Brazilian Air Force. Em *Simpósio de Aplicações Operacionais em Áreas de Defesa (SIGE2023)*, 6. Published, 26 de setembro de 2023. Acesso em 26 de setembro de 2023. https://www.sige.ita.br/edicoes-antiores/2023/st/235455_1.pdf.
- Filho, José Aderson Araújo Passos. 2021. Aprender Simplificar Acelerar: Machine Learning para a Acessibilização de Análises em Escala Urbana. Dissertação de Mestrado. Diss. de mestrado, Universidade Federal de Pernambuco.
- Gobi, Gabriel Henrique, Pedro Lustosa Rege Botelho, Thiago Lobo Ferreira, Thiago Lopes de Araujo e João Paulo de Andrade Dantas. 2023. Ambiente de Simulação Aeroespacial: Capacidades e Potenciais Benefícios para a Indústria de Defesa Brasileira. Em *Congresso Acadêmico sobre Defesa Nacional*. Pirassununga, SP, novembro. <https://doi.org/10.5281/zenodo.10144605>.
- Goodfellow, Ian, Yoshua Bengio e Aaron Courville. 2016. *Deep Learning*. MIT Press. <http://www.deeplearningbook.org>.
- Hastie, Trevor, Robert Tibshirani e Jerome Friedman. 2009. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. 2ª ed. Springer. ISBN: 978-0-387-84857-0.
- Husslage, Bart GM, Gijs Rennen, Edwin R Van Dam e Dick Den Hertog. 2011. Space-filling Latin hypercube designs for computer experiments. *Optimization and Engineering* 12 (4): 611–630.
- James, Gareth, Daniela Witten, Trevor Hastie e Robert Tibshirani. 2013. *An Introduction to Statistical Learning: with Applications in R*. 1ª ed. Springer Texts in Statistics. Springer.
- LeCun, Yann, Yoshua Bengio e Geoffrey Hinton. 2015. Deep Learning. *Nature* 521:436–444.
- McKay, M.D., R.J. Beckman e W.J. Conover. 1979. A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code. *Technometrics* 21 (2): 239–245. <https://doi.org/10.1080/00401706.1979.10489755>.
- Miletto, Marcelo Cogo. 2018. Acelerando uma Aplicação de Simulação Computacional para o Processo de Ablação por Radiofrequência usando GPU. Dissertação de Mestrado. Diss. de mestrado, Universidade Federal do Rio Grande do Sul.
- Milharadas, Marco. 2013. *O consumo de combustível na Força Aérea*. Relatório técnico. Instituto de Estudos Superiores Militares. <http://hdl.handle.net/10400.26/12330>.
- Murphy, Craig A. e T. Perera. 2002. The definition of simulation and its role within an aerospace company. *Simul. Pract. Theory* 9:273–291. [https://doi.org/10.1016/S0928.4869\(01\)00045.3](https://doi.org/10.1016/S0928.4869(01)00045.3).
- Murphy, Kevin P. 2012. *Machine Learning: A Probabilistic Perspective*. The MIT Press.
- Ni, Zhehan. 2022. Reframe the Field of Aerospace Engineering Via Machine Learning: Application and Comparison. *Journal of Physics: Conference Series* 2386. <https://doi.org/10.1088/1742-6596/2386/1/012031>.
- Paula Martins, Anderson Lino de. 2022. Integrando o Machine Learning a um sistema de simulação baseada em agentes. Dissertação de Mestrado. Diss. de mestrado, Pontifícia Universidade Católica do Rio Grande do Sul.
- Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel et al. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12:2825–2830.
- Rocha, Pedro Augusto Ferreira. 2012. Emprego de GPGPUs para Acelerar Simulações do Sistema Humano Inato. Dissertação de Mestrado. Diss. de mestrado, Universidade Federal do Rio Grande do Sul.
- Sutton, Richard S. e Andrew G. Barto. 2018. *Reinforcement Learning: An Introduction*. 2ª ed. Cambridge, MA, USA: MIT Press. ISBN: 978-0-262-03924-6. <http://incompleteideas.net/book/the-book-2nd.html>.

APÊNDICE

REPOSITÓRIO

O repositório online que contém os dados das simulações realizadas no ASA, bem como o código-fonte utilizado no desenvolvimento dos modelos de ML e na geração dos gráficos, pode ser encontrado em <https://github.com/lucasll37/scientific-article-navigation-fighter-plane>.