Trabalho de Inteligência Artificial

Comparação de Algoritmos de Classificação - Avaliação de Jogos na Steam

1. Introdução

Base de Dados

A base utilizada foi extraída do Kaggle - Steam Store Games, contendo dados sobre milhares de jogos publicados na plataforma Steam.

Foram utilizadas as seguintes variáveis:

- price: Preço do jogo
- release_year: Ano de lançamento
- developer: Desenvolvedor
- average_playtime: Tempo médio de jogo
- positive_ratings e negative_ratings: Para calcular uma "nota" dos usuários

Objetivo

O objetivo do trabalho foi classificar os jogos em:

- **Bom**: Se a avaliação dos usuários for maior que 80%
- **Ruim**: Caso contrário

2. Justificativa para Escolha dos Algoritmos

Foram utilizados os seguintes algoritmos clássicos de classificação:

- **Árvore de Decisão**: Interpretação simples e rápida.
- **Random Forest**: Conjunto de árvores, mais robusto e menos propenso a overfitting.
- **SVM (Support Vector Machine)**: Bom para dados com margens bem definidas.
- **Naive Bayes Gaussiano**: Rápido e eficiente para dados numéricos.

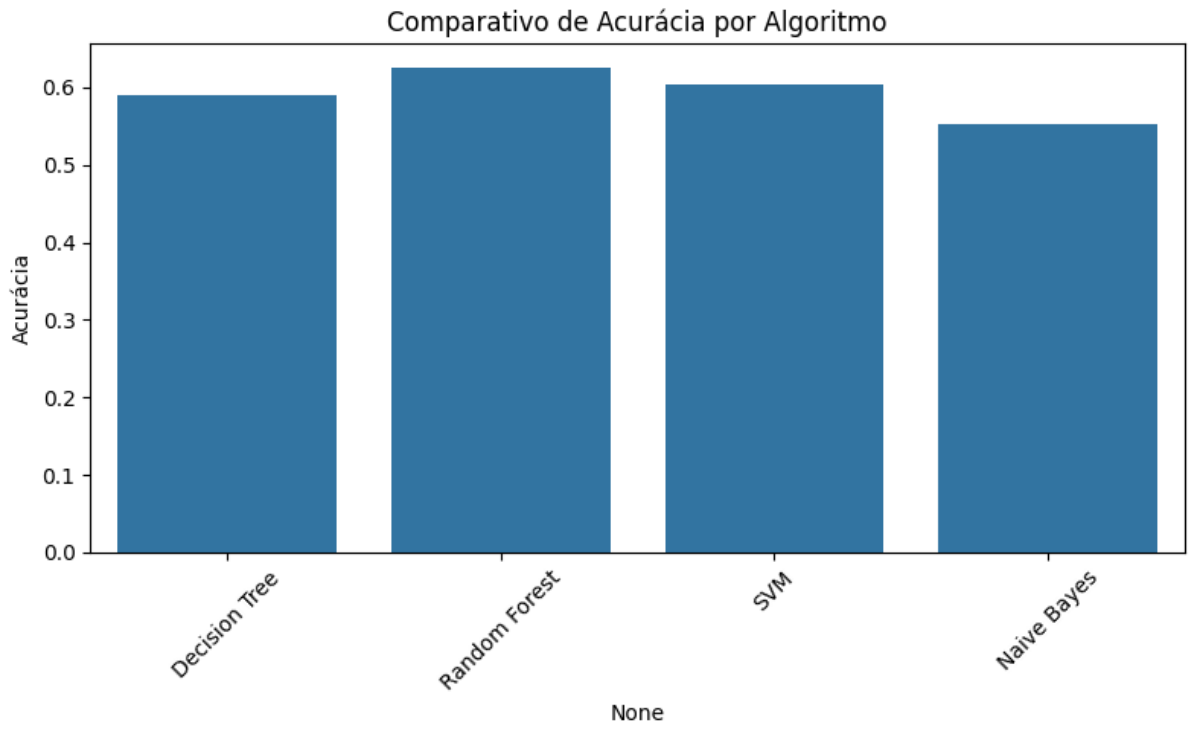
3. Metodologia

- **Pré-processamento**: Tratamento de valores ausentes, extração de release_year, codificação da variável developer, criação da variável-alvo com base nas avaliações.
- **Divisão dos dados**: 80% treino / 20% teste.
- **Métricas utilizadas**:
    - Acurácia
    - Matriz de confusão
    - Precisão, Recall, F1-score

4. Resultados e Comparações

Tabela de Desempenho

Algoritmo	Acurácia	Precisão (Bom)	Recall (Bom)	F1-score (Bom)
Decision Tree	0.5907	0.5294	0.4834	0.5053
Random Forest	0.6261	0.5796	0.4936	0.5331
SVM	0.6040	0.6364	0.1969	0.3008
Naive Bayes	0.5520	0.4774	0.3785	0.4223



5. Discussão

O algoritmo com melhor desempenho geral foi o **Random Forest**, com uma **acurácia de 62,61%** e o melhor equilíbrio entre precisão, recall e F1-score para a classe “Bom”.

- Isso se deve ao fato de que Random Forest é um modelo de conjunto que combina várias árvores de decisão e é menos propenso a overfitting, sendo mais robusto em conjuntos de dados mistos com variáveis categóricas e numéricas.

O **pior desempenho** foi do **Naive Bayes**, com uma acurácia de apenas **55,20%**. Isso provavelmente aconteceu porque esse algoritmo assume que as variáveis são distribuídas normalmente e independentes entre si — o que raramente é verdade nesse tipo de dado real, especialmente com variáveis como price, average_playtime e developer.

Embora o **SVM** tenha tido a maior **precisão** (63,64%) na detecção de jogos bons, seu **recall** foi muito baixo (19,69%), ou seja, ele deixou de identificar muitos jogos bons. Isso pode indicar que ele está sendo conservador demais nas predições positivas.

A **Árvore de Decisão** teve desempenho intermediário, servindo como um bom ponto de comparação por sua simplicidade e interpretabilidade.

6. Conclusão

- Todos os algoritmos apresentaram desempenho razoável, mas alguns foram mais eficazes na identificação de jogos “Bons”.
- O projeto mostrou a importância da escolha correta do algoritmo e do pré-processamento adequado dos dados.
- Trabalhos futuros podem incluir o uso de mais variáveis (como gêneros e tags) ou técnicas de tuning de hiperparâmetros.