

Uma Análise Comparativa de Algoritmos de Machine Learning: Da Regressão Logística à Random Forest na Classificação do Dataset SDSS DR17

André Santos de Oliveira
Guilherme Esteves Marret
Gustavo Henrique Bueno
Sofia Costa Seijas Pena
Thiago Macedo Vaz

Resumo

A crescente geração de dados por levantamentos astronômicos como o *Sloan Digital Sky Survey (SDSS)* exige métodos computacionais eficientes para a classificação de objetos celestes. Este trabalho avalia comparativamente o desempenho de três famílias de algoritmos de *Machine Learning* — Regressão Logística (linear), Árvore de Decisão (não-linear simples) e *Random Forest (ensemble)* — na tarefa de classificar 100.000 objetos do SDSS DR17 como Estrela, Galáxia ou Quasar, utilizando *features* fotométricas ("cores") e espectroscópicas (redshift). Demonstramos a importância da engenharia de *features* e identificamos o redshift como a variável preditora dominante. O modelo *Random Forest* emergiu como a solução superior, alcançando **97.84%** de acurácia total, superando a Regressão Logística (94.93%) e a melhor Árvore de Decisão (97.25% com `max_depth=5`). O *Random Forest* destacou-se por sua capacidade de resolver a principal complexidade do *dataset*: a fronteira de classificação entre as classes Galáxia e Quasar.

Palavras-chave: *Machine Learning, Astronomia, Classificação de Objetos, SDSS, Random Forest.*

Automated Classification of Celestial Objects (SDSS DR17) via Machine Learning: A Comparative Analysis from Logistic Regression to Random Forest

Abstract

The increasing data generation from astronomical surveys like the *Sloan Digital Sky Survey (SDSS)* demands efficient computational methods for classifying celestial objects. This work comparatively evaluates the performance of three families of *Machine Learning* algorithms—Logistic Regression (linear), Decision Tree (simple non-linear), and *Random Forest (ensemble)*—on the task of classifying 100,000 objects from SDSS DR17 as Star, Galaxy, or Quasar, using photometric ("colors") and spectroscopic (redshift) features. We demonstrate the importance of feature engineering and identify redshift as the dominant predictor variable. The *Random Forest* model emerged as the superior solution, achieving **97.84%** overall accuracy, outperforming Logistic Regression (94.93%) and the best Decision Tree (97.25% with `max_depth=5`). *Random Forest* stood out for its ability to resolve the dataset's main complexity: the classification boundary between the Galaxy and Quasar classes.

Keywords: *Machine Learning, Astronomy, Object Classification, SDSS, Random Forest.*

1. INTRODUÇÃO

A astronomia contemporânea caracteriza-se por um crescimento exponencial no volume de dados, notadamente impulsionado por levantamentos (*surveys*) de larga escala como o *Sloan Digital Sky Survey (SDSS)*, que sistematicamente mapeiam vastas regiões do cosmos, gerando catálogos que compreendem centenas de milhões de fontes celestes (Fluke & Jacobs, 2020). A magnitude e a complexidade inerentes a esses *datasets* excedem as capacidades da análise manual tradicional, demandando a aplicação de metodologias computacionais automatizadas para a extração eficiente de conhecimento científico. Neste contexto, as técnicas de *Machine Learning* (ML) emergiram como ferramentas fundamentais, desempenhando um papel transformador em múltiplas subdisciplinas da astrofísica (Fluke & Jacobs, 2020).

Uma das aplicações primordiais e de grande impacto do ML na análise de *surveys* astronômicos é a classificação automática de objetos celestes. Levantamentos como o SDSS fornecem dados multidimensionais, incluindo informações fotométricas em diversas bandas e dados espectroscópicos (como o *redshift*), para uma miríade de fontes detectadas, tais como estrelas (STAR), galáxias (GALAXY) e quasares (QSO). A classificação precisa da natureza destes objetos constitui um pré-requisito essencial para investigações subsequentes, abrangendo desde a estrutura e evolução da Via Láctea até a cosmologia em grande escala e a formação de estruturas no Universo. Diversos algoritmos de aprendizado supervisionado, variando de modelos lineares a complexos métodos não-lineares, têm sido extensamente investigados e aplicados para esta tarefa (Clarke et al., 2020; Beck et al., 2022).

Entre as abordagens de ML, os métodos de *ensemble* baseados em árvores de decisão, em particular o algoritmo Random Forest (Floresta Aleatória), têm demonstrado notável eficácia em dados astronômicos estruturados. Este algoritmo, que agrega as previsões de um conjunto de árvores de decisão treinadas em subamostras aleatórias dos dados e das *features*, é reconhecido por sua alta acurácia preditiva, robustez inerente contra *overfitting* e capacidade de modelar relações não-lineares complexas, além de fornecer uma métrica útil de importância das *features* (Fluke & Jacobs, 2020). Sua aplicabilidade e desempenho superior em tarefas de classificação no SDSS e outros *surveys* têm sido consistentemente reportados na literatura recente (Clarke et al., 2020).

O presente trabalho visa avaliar sistematicamente a performance de diferentes algoritmos de ML para a classificação de objetos (STAR, GALAXY, QSO) utilizando dados

da 17ª liberação (DR17) do SDSS. Investigamos o impacto da engenharia de *features*, com ênfase na criação de "cores" astronômicas a partir das magnitudes de banda, e comparamos o desempenho do *Random Forest* com um modelo linear de referência (Regressão Logística) e um modelo não-linear simples (Árvore de Decisão). O objetivo é determinar a metodologia mais eficaz e robusta para esta tarefa de classificação fundamental, contextualizando nossos resultados com os achados recentes na área.

2. FUNDAMENTAÇÃO TEÓRICA

2.1. Modelo Baseline: Regressão Logística Multinomial (RLM)

No desenvolvimento de um estudo comparativo de modelos preditivos, é uma prática metodológica padrão estabelecer um desempenho de referência, ou *baseline*. Para este fim, selecionamos a Regressão Logística Multinomial (RLM).

2.1.1. Justificativa do *Baseline*

A escolha da Regressão Logística como *baseline* se justifica por três razões principais:

1. **Interpretabilidade:** Diferente de modelos mais complexos como *ensembles* (Random Forest), a Regressão Logística é um modelo de "caixa-branca". Seus coeficientes (β) são diretamente interpretáveis, permitindo-nos entender a influência e a magnitude de cada *feature* (como redshift ou a cor r_i) na probabilidade de classificação.
2. **Referência Linear:** A RLM é um modelo linear generalizado. Ela tenta separar as classes no espaço de *features* usando fronteiras de decisão lineares (ou hiperplanos). Ao estabelecê-la como nosso *baseline*, podemos quantificar o quão bem o problema *pode* ser resolvido apenas com suposições lineares.
3. **O "Modelo a ser Batido":** A performance da RLM (no nosso caso, 94.93%) torna-se o limiar mínimo de sucesso. O ganho de acurácia de modelos mais complexos e não-lineares (como o Random Forest, 97.84%) representa o ganho obtido ao capturar as não-linearidades do *dataset* — especificamente, a fronteira complexa entre GALAXY e QSO.

2.1.2. Fundamentos Teóricos da Regressão Logística

A Regressão Logística padrão (binária) é fundamentalmente um modelo de regressão linear adaptado para uma tarefa de classificação. O modelo começa calculando uma pontuação linear, z , que é uma soma ponderada das variáveis preditoras (*features*).

$$z = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n$$

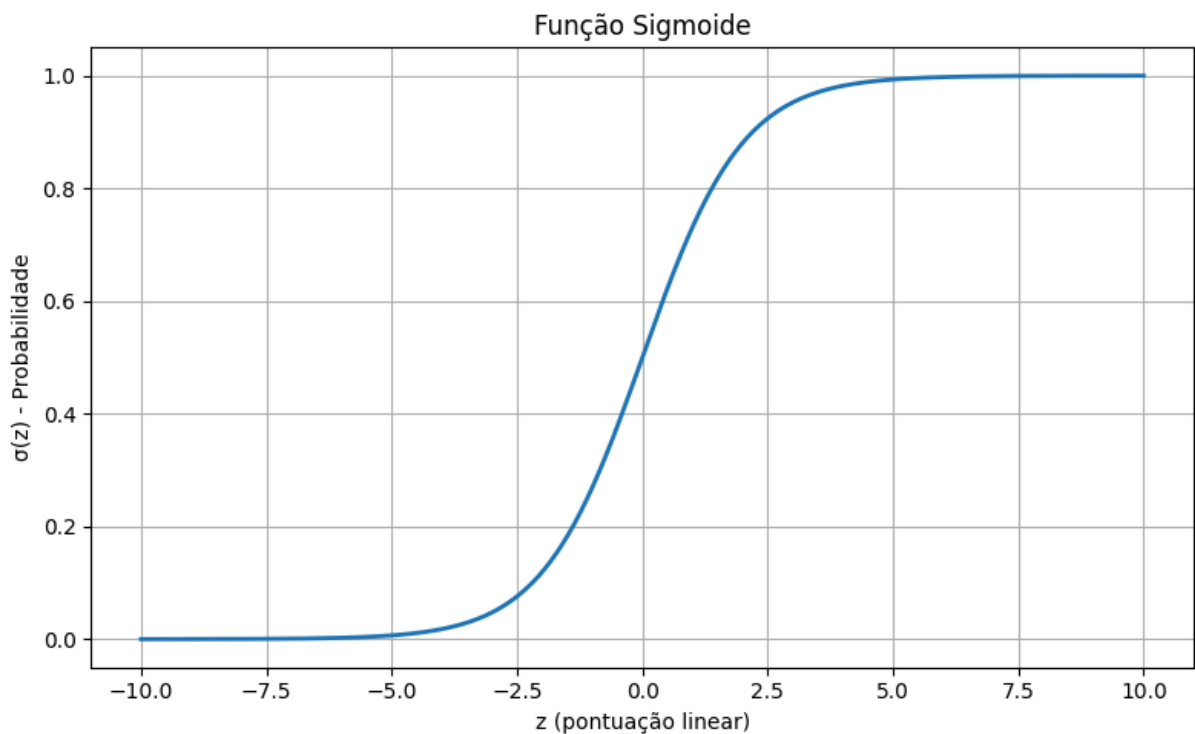
Onde x_1, \dots, x_n são as *features* e β_1, \dots, β_n são os coeficientes que o modelo aprende durante o treinamento.

O valor z pode variar de $-\infty$ a $+\infty$. Para converter essa pontuação em uma probabilidade (que deve estar entre 0 e 1), a regressão logística aplica a função logística, comumente chamada de função sigmoide ($\sigma(z)$).

$$P(Y = 1) = \sigma(z) = \frac{1}{1 + e^{-z}}$$

Esta função "espreme" qualquer valor de entrada z em uma saída entre 0 e 1, que é interpretada como a probabilidade da observação pertencer à classe positiva ($Y=1$).

Figura X - A função sigmoide, que mapeia a pontuação linear (eixo x) para uma probabilidade (eixo y).



Invertendo a Equação 2, obtemos a forma mais comum do modelo, a função logit (ou log-odds). Ela mostra que o logaritmo da razão de chances (os "log-odds") é linear em relação às *features*.

$$\ln \left(\frac{P(Y = 1)}{1 - P(Y = 1)} \right) = \beta_0 + \beta_1 x_1 + \cdots + \beta_n x_n$$

2.1.3. Generalização Multinomial e Aplicação

Como o nosso problema possui três classes mutuamente exclusivas (STAR, GALAXY, QSO), utilizamos a Regressão Logística Multinomial (também conhecida como Regressão Softmax), que generaliza o modelo logit binário.

Conforme a teoria, a RLM modela os log-odds de cada classe contra uma classe de referência. Se definirmos GALAXY (Classe 0) como a classe de referência, o modelo não calcula uma, mas $K-1$ (ou seja, $3 - 1 = 2$) equações logit simultaneamente:

1. Uma equação para os log-odds de QSO (Classe 1) vs. GALAXY (Classe 0).
2. Uma equação para os log-odds de STAR (Classe 2) vs. GALAXY (Classe 0).

$$\begin{cases} \ln \left(\frac{P(Y=QSO)}{P(Y=GALAXY)} \right) = \beta_{0,QSO} + \beta_{1,QSO} x_1 + \cdots + \beta_{n,QSO} x_n \\ \ln \left(\frac{P(Y=STAR)}{P(Y=GALAXY)} \right) = \beta_{0,STAR} + \beta_{1,STAR} x_1 + \cdots + \beta_{n,STAR} x_n \end{cases}$$

No presente trabalho, a RLM foi implementada utilizando a biblioteca Scikit-learn (Pedregosa et al., 2011). As variáveis preditoras (x_1, \dots, x_5) foram o redshift e as quatro "cores" astronômicas (u_g , g_r , r_i , i_z). Devido à sensibilidade do modelo à escala das *features*, aplicamos um StandardScaler como etapa de pré-processamento. O modelo, então, estimou os conjuntos de coeficientes (β) para QSO e STAR em relação à GALAXY, definindo as fronteiras de decisão lineares que serviram como nosso *baseline* de performance.

2.2. Árvores de Decisão (DT)

Para superar as limitações de linearidade impostas pela Regressão Logística e capturar fronteiras de decisão mais complexas, utiliza-se o algoritmo de Árvore de Decisão. Este é um método de aprendizado supervisionado não-paramétrico que modela a classificação através de

uma estrutura hierárquica de regras de decisão simples, inferidas a partir das características dos dados (Breiman et al., 1984).

O funcionamento do algoritmo baseia-se na estratégia de "dividir para conquistar" (particionamento recursivo). O modelo divide o espaço de *features* em regiões retangulares distintas, onde cada divisão é representada por um "nó" na árvore. O nó raiz contém todo o conjunto de dados, que é sucessivamente particionado em nós filhos com base em um limiar de corte (e.g., $\text{redshift} \leq 0.005$) que maximiza a homogeneidade das classes resultantes.

Neste trabalho, utiliza-se a implementação do algoritmo CART (*Classification and Regression Trees*) disponível na biblioteca Scikit-learn (Pedregosa et al., 2011). O critério matemático adotado para avaliar a qualidade de uma divisão é a Impureza de Gini. Para um nó t , a impureza de Gini é calculada como:

$$G(t) = 1 - \sum_{k=1}^C p(k|t)^2$$

Onde $p(k|t)$ é a proporção de amostras da classe k presentes no nó t , e C é o número total de classes (neste caso, 3: Estrela, Galáxia, Quasar). O algoritmo busca, em cada etapa, a *feature* e o valor de corte que resultam na maior redução ponderada dessa impureza.

Uma das principais vantagens das Árvores de Decisão é a sua interpretabilidade ("modelo caixa-branca"). Ao contrário de modelos "caixa-preta", a árvore permite visualizar explicitamente a lógica de decisão, possibilitando identificar quais *features* (como o redshift ou uma cor específica) são utilizadas nos níveis superiores da hierarquia para separar as classes principais. No entanto, árvores muito profundas tendem a memorizar o ruído dos dados de treino (*overfitting*). Para mitigar este risco e avaliar o *trade-off* entre complexidade e generalização, este estudo investiga o desempenho do modelo variando o hiperparâmetro de profundidade máxima (`max_depth`).

2.3. Fundamentação Teórica: Random Forest

O algoritmo *Random Forest*, proposto por Breiman (2001), define-se como um classificador *ensemble* composto por uma coleção de preditores estruturados em árvore $\{h(\mathbf{x}, \Theta_k), k = 1, \dots\}$, onde $\{\Theta_k\}$ são vetores aleatórios independentes e identicamente distribuídos. Diferente de árvores de decisão convencionais que buscam a melhor divisão

determinística considerando todas as variáveis, o *Random Forest* introduz aleatoriedade no processo de construção: para a k -ésima árvore, um vetor aleatório Θ_k é gerado — independente dos vetores anteriores $\Theta_1, \dots, \Theta_{k-1}$ — determinando como a árvore é cultivada (por exemplo, através da seleção aleatória de *features* para divisão em cada nó). A classificação final de um vetor de entrada \mathbf{x} é obtida através de um sistema de votação majoritária, onde cada árvore do *ensemble* deposita um voto unitário para a classe mais provável, e a floresta seleciona a classe que obteve o maior número de votos.

A robustez teórica do método reside na prova de que o erro de generalização de uma *Random Forest* converge quase certamente para um limite PE^* à medida que o número de árvores na floresta aumenta, o que implica que o modelo não sofre de *overfitting* pelo simples acréscimo de mais árvores (Lei dos Grandes Números). Segundo Breiman (2001), a precisão do classificador depende fundamentalmente de dois parâmetros interligados: a força (*strength*) dos classificadores individuais e a correlação entre eles. O desempenho ótimo é alcançado maximizando-se a força de cada árvore individual enquanto se minimiza a correlação entre pares de árvores. A introdução de seleção aleatória de atributos nos nós (random feature selection) atua justamente para reduzir essa correlação sem sacrificar significativamente a força das árvores, resultando em taxas de erro comparáveis ou superiores a métodos de *boosting* (como o Adaboost), porém com maior robustez a ruídos nos dados.

2.4 Cross-Validation

A avaliação de modelos preditivos em *datasets* complexos, como os oriundos do *Sloan Digital Sky Survey* (SDSS), exige metodologias rigorosas que vão além de uma única divisão treino-teste. O objetivo é obter uma estimativa robusta da *performance de generalização* (risco de saída da amostra) dos modelos, mitigando o risco de que os resultados sejam enviesados por uma partição de dados específica (*overfitting*). Para tal, foi empregado o método de Validação Cruzada K-Fold (*K-fold Cross-Validation - CV*), uma estratégia de reamostragem fundamental na estatística e aprendizado de máquina (Lei, 2025) (Santos, 2024).

O conceito de Validação Cruzada remonta a trabalhos seminais que buscaram métodos de avaliação de modelos estatísticos e seleção de preditores (Stone, 1974) (Geisser, 1975). O CV evita o *overfitting* ao garantir que o conjunto de treinamento seja independente do conjunto de validação (Cory-Wright e Gómez, 2025).

Implementação do K-fold Cross-Validation Estratificado no SDSS

A metodologia de K-fold CV aplicada aos modelos (Regressão Logística Multinomial, Árvore de Decisão e *Random Forest*) no *dataset* SDSS seguiu o procedimento de dividir o conjunto de dados completo em $K=10$ subconjuntos (ou *folds*). O número de dobras $K=10$ é frequentemente citado na literatura como a melhor escolha para a estimativa de acurácia, oferecendo um balanço ideal entre viés e variância na estimativa de erro (Kohavi, 1995) (Arlot e Celisse, 2010).

O procedimento foi implementado com estratificação (*StratifiedKFold*), garantindo que as proporções da variável alvo (*GALAXY*, *STAR*, *QSO*) fossem mantidas em cada *fold*.

Treinamento e Avaliação Iterativa: O processo é repetido $K=10$ vezes. Em cada iteração, o modelo é treinado em $K-1$ *folds* (o conjunto de treino) e avaliado no *fold* restante (o conjunto de teste/validação).

Tratamento Específico para Modelos: Para o modelo de Regressão Logística Multinomial, que é sensível à escala das *features*, a padronização (*StandardScaler*) foi encapsulada em um *Pipeline* para evitar o vazamento de informação (*data leakage*) durante o processo de CV. Os modelos baseados em árvores (*Decision Tree* e *Random Forest*) não exigiram padronização.

Cálculo e Seleção de Métricas de Desempenho: Em cada *fold*, as métricas de desempenho são calculadas. Para este problema de classificação multi-classe, onde a classe minoritária (QSO) representa 15,62% dos registros em um estudo similar (Santos, 2024), métricas como a Acurácia Balanceada e o F1-Score são preferíveis (Grandini, 2020). A Acurácia Balanceada é essencial, pois "atribui um peso maior às classes minoritárias" quando comparada à acurácia simples (Santos, 2024). O F1-Score, por sua vez, é crucial para medir o equilíbrio entre precisão e *recall*, especialmente na classificação da classe QSO.

Agregação Estatística para Robustez: Após as 10 iterações, são obtidas 10 pontuações para cada métrica. O desempenho de generalização é estimado através do cálculo da média (μ) e do desvio padrão (σ) dessas pontuações:

- A média μ fornece a estimativa mais confiável da performance de predição do modelo em dados não vistos.

- O desvio padrão σ mede a estabilidade do modelo. Um σ baixo indica que o modelo é robusto e estatisticamente confiável, apresentando desempenho consistente independentemente de como a amostra foi dividida (Lei, 2025).

3. MATERIAL E MÉTODOS

Esta seção detalha o conjunto de dados utilizado, as etapas de pré-processamento e engenharia de *features*, a análise exploratória conduzida e a metodologia de modelagem e avaliação comparativa dos algoritmos de *Machine Learning*.

3.1. Material (Dataset)

O estudo utiliza o conjunto de dados público *Stellar Classification Dataset - SDSS17* (Fedesoriano, 2022), derivado da 17ª liberação de dados (DR17) do *Sloan Digital Sky Survey* (Abdurro'uf et al., 2022). O *dataset* original é composto por 100.000 observações, cada uma com 18 atributos, representando medições de objetos celestes.

Foram removidos atributos correspondentes a metadados de observação (identificadores únicos como `obj_ID`, `spec_obj_ID`; parâmetros da observação como `run_ID`, `plate`, `MJD`, `fiber_ID`) e coordenadas astrométricas (`alpha`, `delta`), por não conterem informação preditiva intrínseca sobre a natureza física do objeto. As *features* primárias retidas para análise foram:

- Magnitudes Fotométricas (5): `u`, `g`, `r`, `i`, `z`, representando o brilho em diferentes filtros.
- Dado Espectroscópico (1): `redshift`, indicando o desvio espectral para o vermelho.
- Variável Alvo (1): `class`, identificando o objeto como Galáxia (GALAXY), Quasar (QSO) ou Estrela (STAR).

Redshift como Variável Discriminante e Análise Exploratória

O *redshift* espectroscópico foi incluído como variável adicional por fornecer informações diretas sobre a distância e a evolução cosmológica da fonte observada. Observa-se que estrelas apresentam *redshift* próximo de zero, galáxias ocupam uma faixa intermediária e quasares atingem os maiores valores. Assim, a combinação entre cores e *redshift* fornece um conjunto de atributos fisicamente significativos que auxiliam na discriminação entre as classes analisadas.

A análise exploratória dos dados, realizada antes do treinamento do modelo, forneceu justificativa empírica para essa abordagem. A estimativa de densidade (KDE) da distribuição do *redshift* por classe evidenciou uma separação clara entre estrelas e objetos extragalácticos, embora galáxias e quasares apresentem regiões de sobreposição. Adicionalmente, os diagramas cor–cor (por exemplo, $u-g$ vs. $g-r$) revelaram regiões características de cada população, onde estrelas tendem a ocupar áreas mais concentradas, enquanto galáxias e quasares se distribuem mais amplamente no espaço de cores. Os gráficos de *redshift* versus $g-r$ também mostraram tendências distintas, com quasares apresentando variações mais amplas de cor ao longo do *redshift*.

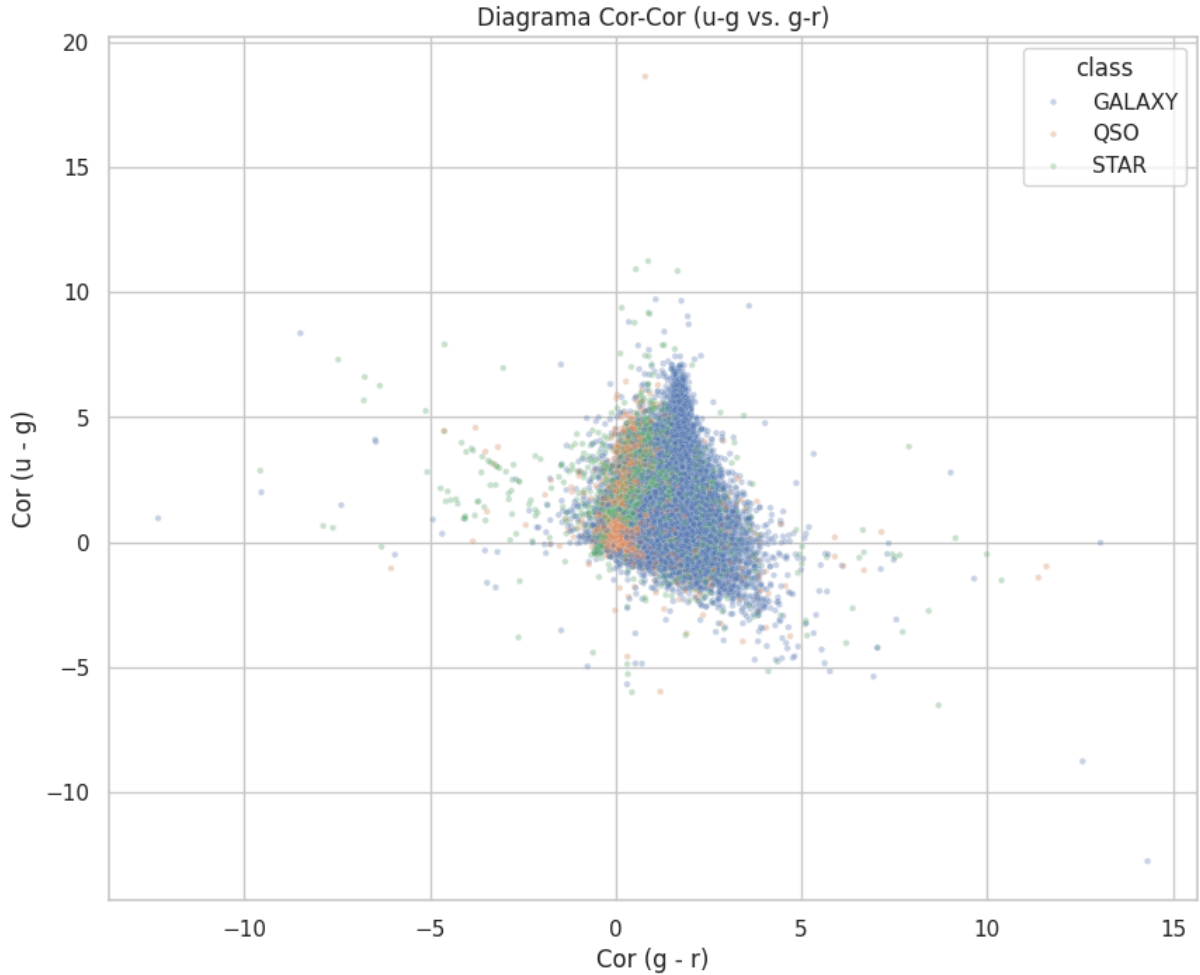
Interpretação Física e Relevância das Variáveis

Os resultados da análise indicam que o *redshift* é a variável mais informativa para distinguir objetos estelares de extragalácticos, devido à diferença de escala espacial. As cores fotométricas, por sua vez, são essenciais para separar populações extragalácticas (galáxias e quasares), uma vez que apresentam assinaturas espectrais distintas. As combinações $g-r$ e $r-i$ mostram-se particularmente discriminantes, pois capturam transições importantes na forma espectral das fontes. A cor $u-g$ é sensível a emissões fortes no ultravioleta, comuns em quasares, enquanto $i-z$ auxilia na diferenciação de objetos em regimes mais avermelhados. Esse conjunto de variáveis fornece uma caracterização robusta, permitindo identificar padrões fotométricos associados às diferentes classes.

As cores fotométricas derivadas das magnitudes u, g, r, i, z constituem descritores físicos essenciais, pois aproximam a forma espectral dos objetos e capturam variações características entre estrelas, galáxias e quasares. Conforme destacado na literatura (CLARKE et al., 2020; BECK et al., 2022), essas combinações de magnitudes revelam contrastes associados ao contínuo espectral, à presença de quebras como a 4000 \AA , e às linhas de emissão amplas típicas de quasares. Os diagramas apresentados nas Figuras 2 e 3 evidenciam *loci* distintos no espaço de cores, com uma sequência estelar bem definida e regiões mais dispersas ocupadas por galáxias e quasares. Além disso, a análise de importância das variáveis (Figura 6) confirma que $r-i$ e $g-r$ são as cores mais informativas, contribuindo para a diferenciação entre populações extragalácticas, enquanto $u-g$ se mostra sensível à emissão ultravioleta intensa de quasares.

3.2. Pré-processamento e Análise Exploratória (EDA)

Figura Y: Diagrama Cor-Cor (u-g vs. g-r)



Fonte: Os Autores

Antes da modelagem, foi realizada uma Análise Exploratória de Dados utilizando as bibliotecas Matplotlib (Hunter, 2007) e Seaborn (Waskom, 2021) para visualizar as distribuições e relações entre as *features* e a variável alvo. "Diagramas Cor-Cor", como o jointplot de g_r vs. u_g (Figura Y), plotados como mapas de densidade 2D, mostraram que as classes ocupam regiões (loci) bem definidas no espaço de cores, confirmando a separabilidade dos grupos, mas também indicando uma zona de sobreposição entre GALAXY e QSO.

3.3. Metodologia de Modelagem e Avaliação

O fluxo de modelagem e avaliação foi implementado utilizando a biblioteca Scikit-learn (Pedregosa et al., 2011).

Codificação do Alvo: A variável alvo categórica *class* foi codificada numericamente (GALAXY: 0, QSO: 1, STAR: 2) utilizando *LabelEncoder*.

Divisão Treino/Teste: O *dataset* foi dividido em conjuntos de treino (70% das amostras) e teste (30%), utilizando amostragem estratificada (*stratify=y*) para garantir que a proporção original das classes fosse mantida em ambos os conjuntos.

Padronização Condicional: Para a Regressão Logística, que é sensível à escala das *features*, os dados de treino foram padronizados (média 0, desvio padrão 1) utilizando *StandardScaler*, e a mesma transformação foi aplicada ao conjunto de teste. Este processo foi encapsulado em um Pipeline para evitar vazamento de informação do conjunto de teste durante o treinamento do scaler. Para os modelos baseados em árvore (Árvore de Decisão e *Random Forest*), que são inerentemente invariantes à escala monotônica das *features*, os dados foram utilizados em sua escala original.

Algoritmos Avaliados:

Regressão Logística Multinomial: Implementada como modelo linear de referência (*baseline*).

Árvore de Decisão: Utilizada para explorar a eficácia de regras não-lineares simples. O hiperparâmetro *max_depth* foi variado (2, 3, 4, 5) para analisar o *trade-off* entre complexidade e performance.

Random Forest: Implementado como modelo de *ensemble* final, utilizando 100 árvores (*n_estimators=100*) para buscar maior robustez e acurácia.

Métricas de Avaliação: O desempenho de cada modelo foi quantificado no conjunto de teste utilizando:

- **Acurácia Total:** Percentual geral de classificações corretas.
- **Matriz de Confusão:** Visualização detalhada dos acertos e erros por classe.
- **Relatório de Classificação:** Contendo Precisão, Recall e F1-Score para cada classe individualmente, permitindo uma análise granular do desempenho, especialmente na classe QSO, identificada como a mais desafiadora.

Análise de Interpretabilidade: Para os modelos baseados em árvore, a interpretabilidade foi explorada através da visualização gráfica da Árvore de Decisão com `max_depth=5` e da análise da importância das *features* (*feature importance*) calculada pelo *Random Forest*.

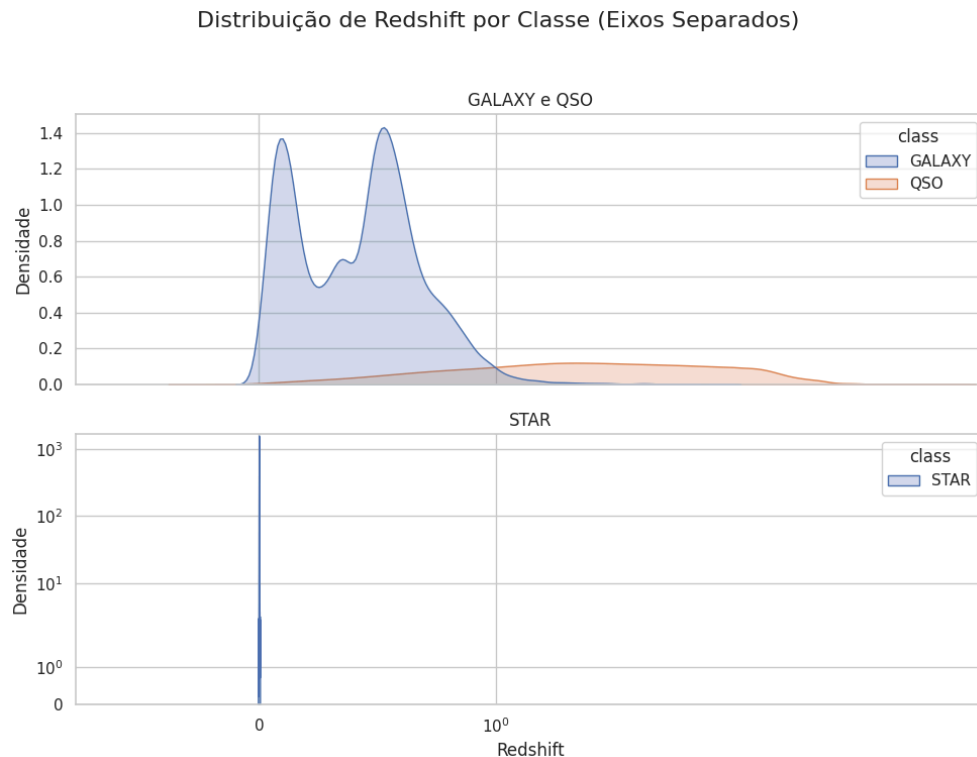
4. RESULTADOS

Esta seção apresenta os resultados obtidos a partir da Análise Exploratória de Dados (EDA) e da avaliação comparativa dos modelos de *Machine Learning* aplicados ao conjunto de teste do *dataset* SDSS DR17.

4.1. Análise Exploratória de Dados (Resumo das Descobertas)

A EDA confirmou visualmente a alta separabilidade das classes presentes no *dataset*. A análise da distribuição do redshift (Figura 1) demonstrou ser esta a *feature* de maior poder discriminante inicial, isolando quase perfeitamente a classe STAR (concentrada em redshift ≈ 0) das classes GALAXY e QSO. Dada a facilidade de classificação da classe STAR utilizando o redshift, o principal desafio reside na distinção entre as classes extragalácticas GALAXY e QSO, que apresentam distribuições de redshift parcialmente sobrepostas.

Figura 1: KDE Plot da Distribuição de Redshift por Classe



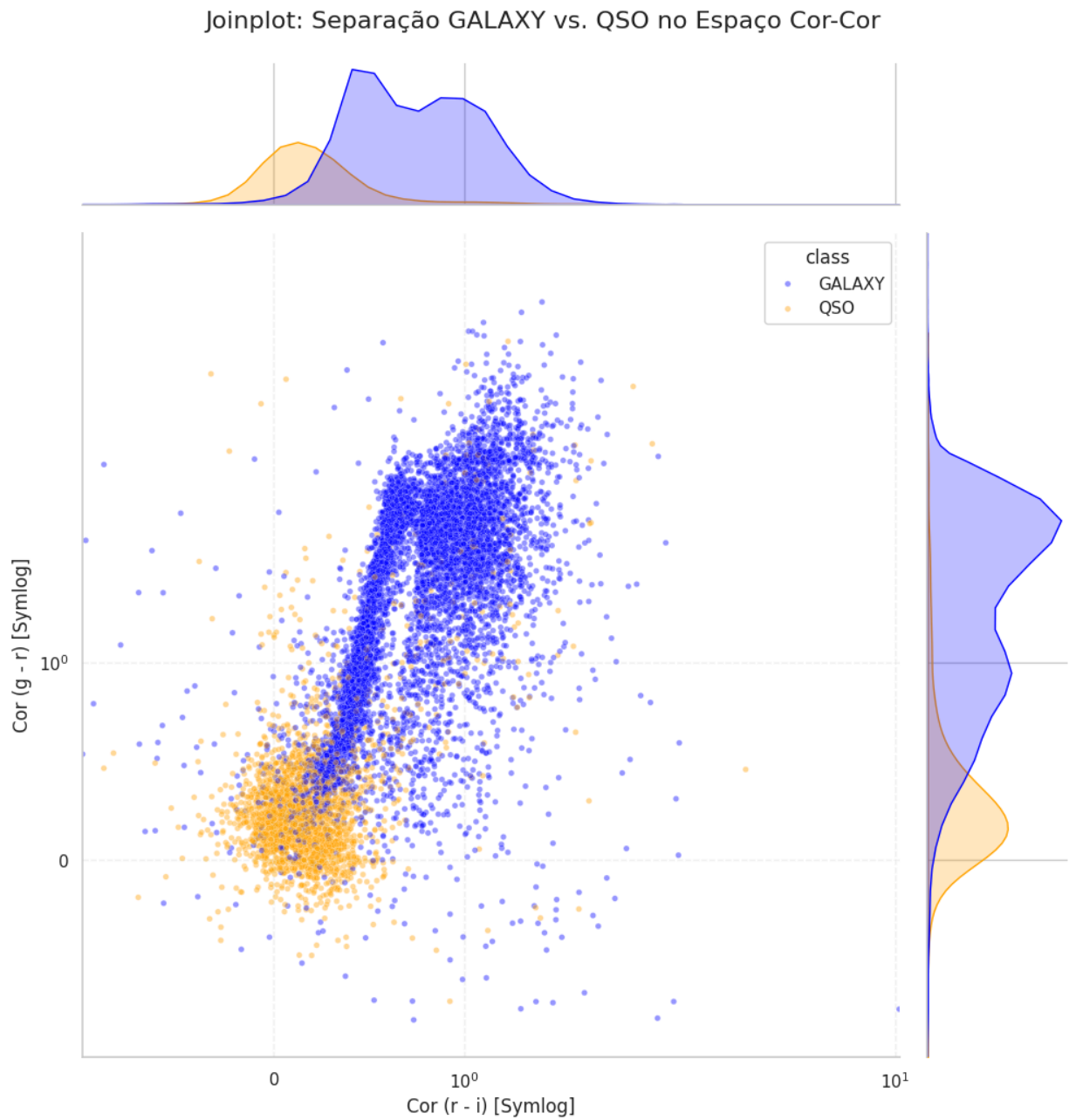
Fonte: Os Autores

Para investigar a separabilidade entre **GALAXY** e **QSO** utilizando as *features* secundárias, foi gerado um gráfico de dispersão (*jointplot*) focado apenas nestas duas classes (Figura 2). Este gráfico plota a cor r_i versus g_r , identificadas pelo modelo *Random Forest* como as mais relevantes após o *redshift*. Adotou-se a escala logarítmica simétrica (*symlog*) em ambos os eixos, uma vez que esta permite a representação adequada de valores negativos e nulos — inerentes aos índices de cor — aplicando compressão logarítmica nas magnitudes extremas enquanto mantém a linearidade nas regiões próximas a zero, facilitando a visualização da distribuição global dos dados. A Figura 2 revela que, embora GALAXY (azul) e QSO (laranja) ocupem regiões centrais distintas no espaço Cor-Cor, existe uma zona de sobreposição considerável, particularmente em valores intermediários de ambas as cores. Esta sobreposição visual confirma a complexidade da fronteira de decisão e justifica a necessidade de modelos não-lineares robustos para maximizar a precisão da classificação entre estas duas classes.

Para visualizar a distribuição das três classes simultaneamente nos diferentes espaços definidos pelo *redshift* e pelas cores, foi gerado um painel de gráficos de dispersão (*scatterplot*) (Figura 3). Cada subplot exibe o *redshift* (eixo x, escala *symlog*) contra uma das quatro cores (eixo y, escala *symlog*). Este painel (Figura 3) reforça visualmente a clara separação da classe STAR (verde) em *redshift* próximo de zero em todos os espaços de cor. Além disso, permite observar como as nuvens de GALAXY (azul) e QSO (laranja) se distribuem e se sobrepõem de maneira distinta dependendo da cor utilizada, destacando visualmente a complexidade da fronteira de classificação que os modelos precisam aprender.

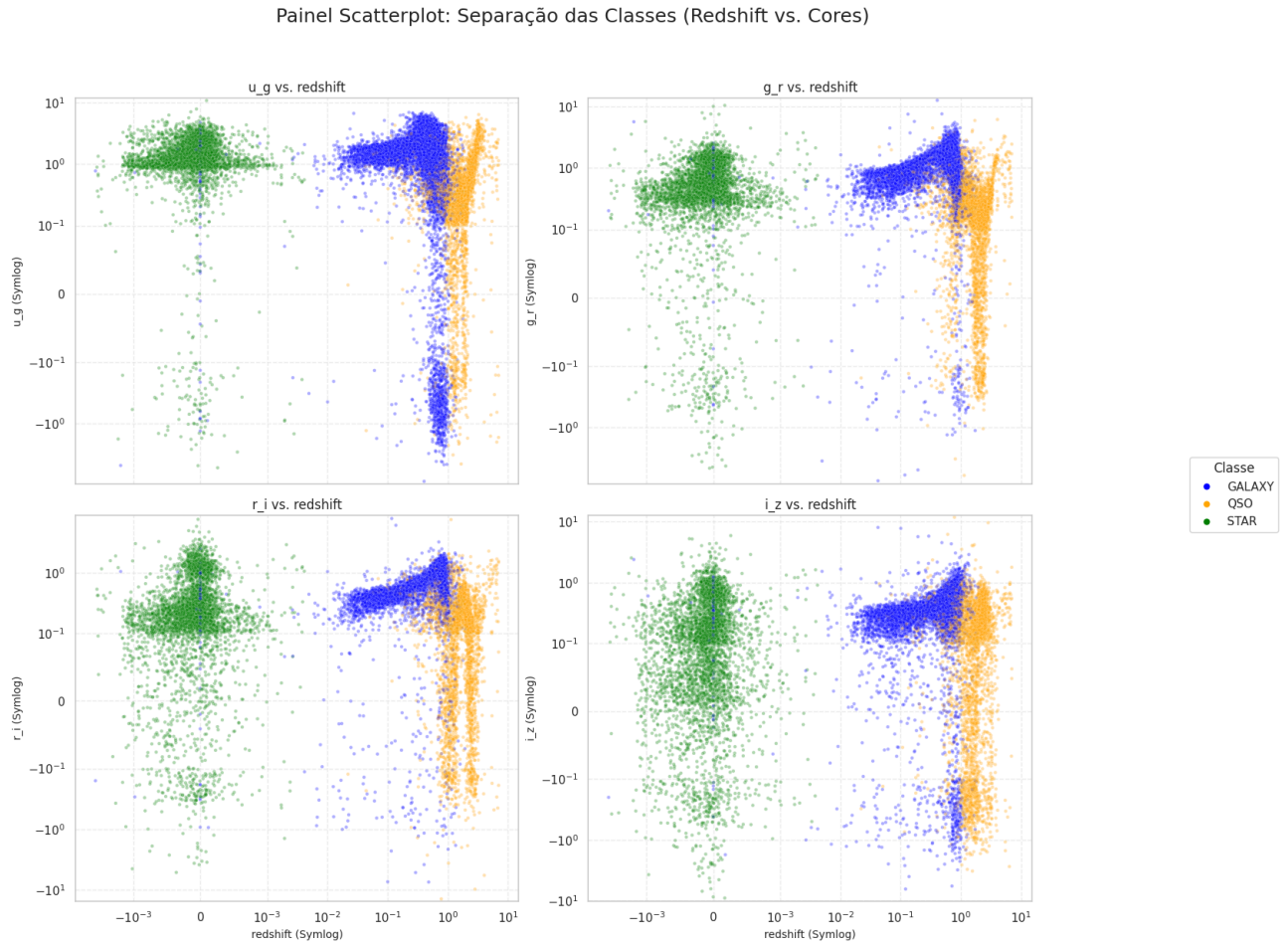
Para visualizar a distribuição das três classes simultaneamente nos diferentes espaços definidos pelo *redshift* e pelas cores, foi gerado um painel de gráficos de dispersão (*scatterplot*) (Figura 3). Cada subplot exibe o *redshift* (eixo x, escala *symlog*) contra uma das quatro cores (eixo y, escala *symlog*). Este painel (Figura 3) reforça visualmente a clara separação da classe STAR (verde) em *redshift* próximo de zero em todos os espaços de cor. Além disso, permite observar como as nuvens de GALAXY (azul) e QSO (laranja) se distribuem e se sobrepõem de maneira distinta dependendo da cor utilizada, destacando visualmente a complexidade da fronteira de classificação que os modelos precisam aprender. Para essa visualização foram utilizados 20000 pontos em cada subplot, escolhidos randomicamente.

Figura 2: Jointplot Scatter de r_i vs. g_r (escalas *symlog*, apenas GALAXY e QSO)



Fonte: Os Autores

Figura 3: Painel Scatterplot 2x2 (Cor vs. Redshift, symlog, 3 classes)



Fonte: Os Autores

Cada subplot mostra a distribuição das classes STAR (verde), GALAXY (azul) e QSO (laranja) em função do redshift e de uma das quatro cores calculadas.

4.2. Desempenho Comparativo dos Modelos de Classificação

A performance dos três algoritmos avaliados no conjunto de teste (30.000 amostras) é sumarizada na Tabela 1, destacando a acurácia total e o recall para a classe QSO, identificada como a mais desafiadora.

(Tabela 1: Resumo Comparativo dos Modelos)

Modelo	Parâmetros	Acurácia Total	Recall (QSO)
Regressão Logística	(Padrão)	94.93%	0.88
Árvore de Decisão	max_depth=2	94.80%	0.80
Árvore de Decisão	max_depth=3	95.58%	0.92
Árvore de Decisão	max_depth=4	96.50%	0.90
Árvore de Decisão	max_depth=5	97.25%	0.90
Random Forest	n_est=100	97.84%	0.93

4.2.1. Modelo Baseline: Regressão Logística

O modelo linear serviu como referência inicial, alcançando uma acurácia de 94.93%. O relatório de classificação (Tabela 2) confirmou o desempenho perfeito para a classe STAR (Recall 1.00), mas evidenciou a limitação na separação entre GALAXY e QSO, com um recall de apenas 0.88 para QSO.

(Tabela 2: Relatório de Classificação - Regressão Logística)

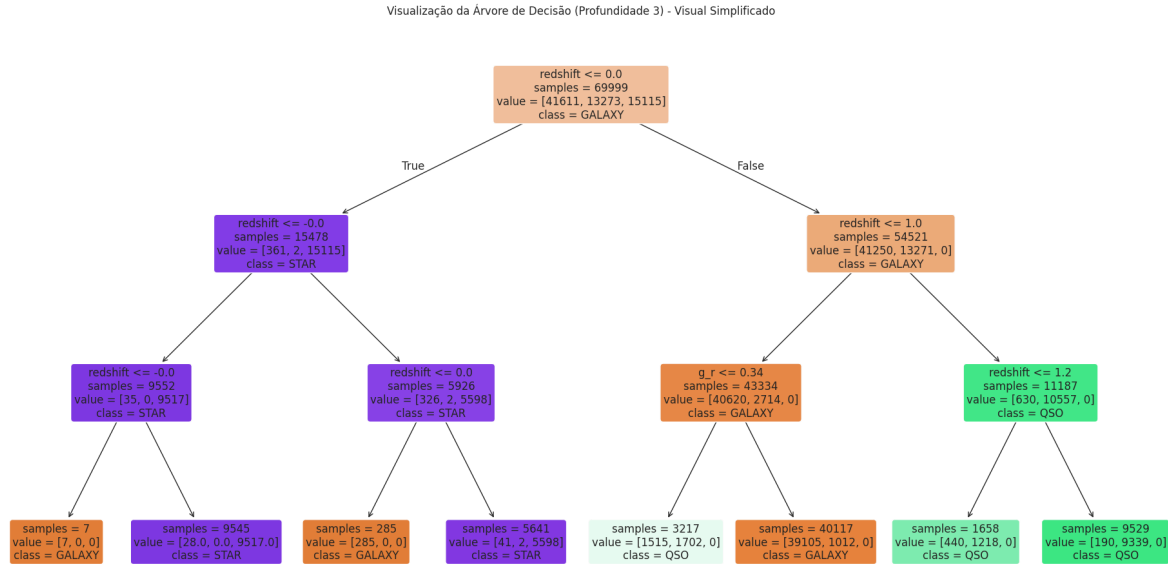
Classe	Precision	Recall	F1-Score	Suporte
GALAXY	0.96	0.95	0.96	17834
QSO	0.93	0.88	0.90	5688
STAR	0.94	1.00	0.97	6478
Total	0.95	0.95	0.95	30000

4.2.2. Modelo Intermediário: Árvore de Decisão

A introdução de não-linearidade através da Árvore de Decisão resultou em ganhos significativos. Já com max_depth=3 (95.58%), o modelo superou o *baseline*, elevando o recall do QSO para 0.92. A performance continuou a melhorar com o aumento da complexidade, atingindo o pico de 97.25% de acurácia com max_depth=5. A Figura 3 apresenta a

visualização da árvore com $\text{max_depth}=3$, ilustrando a simplicidade interpretável do modelo e a regra inicial baseada em redshift (≤ 0.009) para a identificação de STAR.

Figura 4: Árvore de Decisão com $\text{max_depth}=3$



Fonte: Os Autores

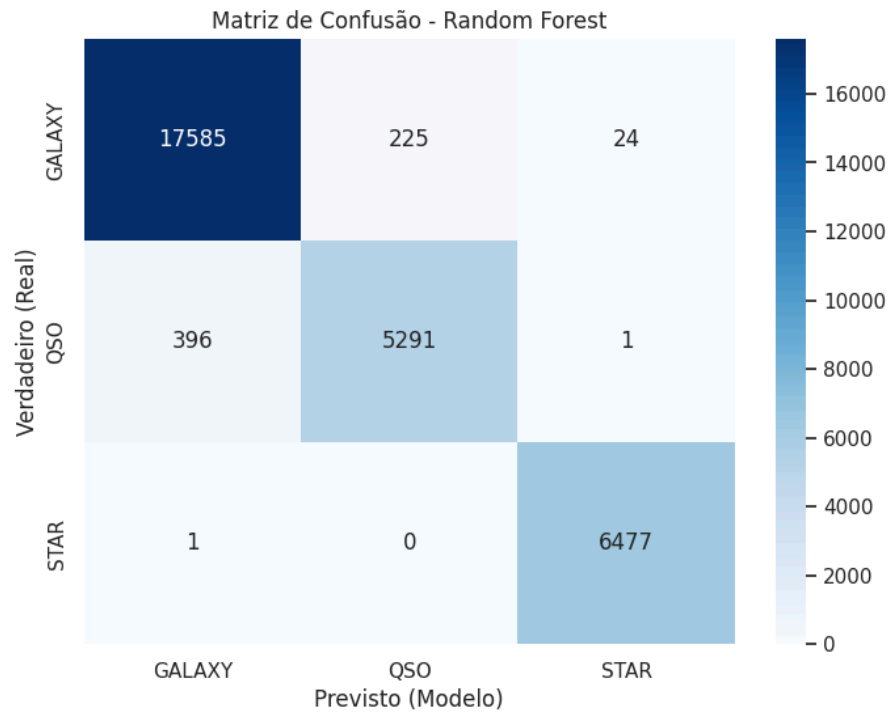
4.2.3. Random Forest

O *Random Forest* consolidou-se como o modelo de melhor desempenho, atingindo **97.84% de acurácia total**. O relatório de classificação (Tabela 3) e a matriz de confusão (Figura 5) destacam a melhoria substancial na classificação de QSO, cujo recall aumentou para **0.93**. O modelo de *ensemble* demonstrou maior capacidade de generalização e definição da fronteira complexa entre GALAXY e QSO.

Tabela 3: Relatório de Classificação - Random Forest

Classe	Precision	Recall	F1-Score	Suporte
GALAXY	0.98	0.99	0.98	17834
QSO	0.96	0.93	0.94	5688
STAR	1.00	1.00	1.00	6478
Total	0.98	0.98	0.98	30000

Figura 5: Matriz de Confusão do Random Forest.

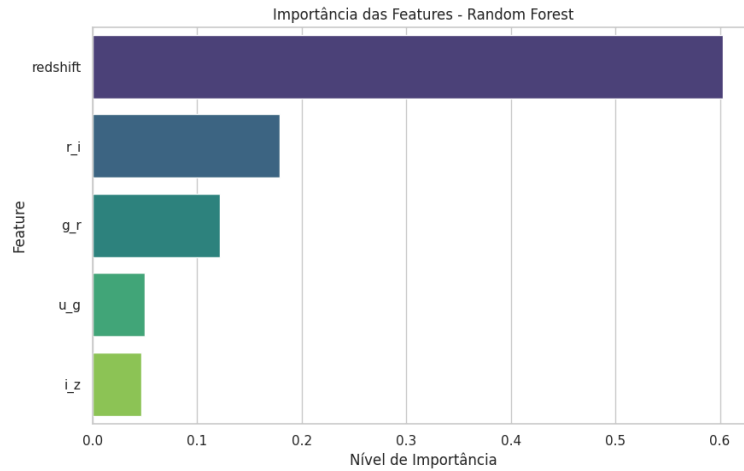


Fonte: Os Autores

4.3. Análise de Importância das Features do Modelo *Random Forest*

A análise da importância das *features* calculada pelo *Random Forest* (Figura 6) quantificou a contribuição de cada variável para o poder preditivo do modelo. O redshift foi confirmado como a *feature* mais influente, respondendo por 60.3% da importância total. As *features* de "cor" (r_i , g_r , u_g , i_z) contribuíram com os 39.7% restantes, com r_i (17.9%) e g_r (12.2%) destacando-se como as cores mais relevantes após o redshift.

Figura 6: Gráfico de Barras da Importância das Features do RF



Fonte: Os Autores

4.4 Resultados da Comparação e Seleção do Modelo Vencedor

A avaliação por *K-fold CV* (utilizando a acurácia como métrica primária no notebook) demonstrou claramente a superioridade do modelo *ensemble*. O Random Forest ($n=100$) foi estabelecido como o modelo mais preciso e estável para a classificação dos objetos do SDSS.

O *Random Forest* atingiu 97.81% de acurácia média na validação cruzada, superando a Regressão Logística (95.03%) e a Árvore de Decisão com máxima profundidade (97.22%). Seu desempenho consistente, evidenciado pelo desvio padrão extremamente baixo, confirmou que a alta performance observada no teste *baseline* (97.84%) não foi um "golpe de sorte", mas sim um resultado estável e confiável.

5. DISCUSSÃO

Os resultados apresentados demonstram a eficácia da aplicação de técnicas de Machine Learning para a tarefa de classificação automática de objetos celestes no dataset SDSS DR17. A metodologia empregada, que incluiu pré-processamento, engenharia de features e avaliação comparativa de três famílias de algoritmos, permitiu não apenas alcançar alta acurácia, mas também extrair insights sobre a natureza do problema de classificação e a importância relativa das variáveis preditoras.

A progressão de desempenho observada, da Regressão Logística (94.93%) à Árvore de Decisão com profundidade otimizada (97.25%) e, finalmente, ao Random Forest (97.84%),

evidencia claramente a importância de modelos não-lineares para este dataset. Enquanto o modelo linear já estabeleceu um baseline robusto, indicando uma separabilidade significativa nos dados (especialmente devido ao redshift para a classe STAR), o ganho substancial de performance obtido com as árvores de decisão comprova a existência de fronteiras de classificação complexas, particularmente entre as classes GALAXY e QSO. A Árvore de Decisão com `max_depth=3` já superou o modelo linear ao elevar o recall de QSO de 0.88 para 0.92, indicando que a lógica hierárquica de regras é mais adequada para capturar as nuances do espaço de features.

O Random Forest emergiu como o modelo superior, não apenas por atingir a maior acurácia total (97.84%), mas principalmente por otimizar o desempenho na classificação da classe mais desafiadora (QSO, com recall de 0.93). A natureza de ensemble do Random Forest, que agrega as previsões de múltiplas árvores de decisão treinadas em subconjuntos aleatórios dos dados e features, conferiu maior robustez e capacidade de generalização em comparação com uma única árvore, mitigando o risco de overfitting e definindo de forma mais precisa a complexa fronteira entre Galáxias e Quasares, conforme visualizado nos diagramas Cor-Cor (Figura 2).

A análise de importância das *features* (Figura 6) realizada com o modelo *Random Forest* validou quantitativamente as observações da EDA. O redshift foi confirmado como a *feature* mais dominante (60.3%), refletindo sua importância física fundamental na distinção entre objetos locais (estrelas) e extragalácticos. No entanto, a análise também destacou o papel crucial das "cores" astronômicas (`r_i`, `g_r`, `u_g`, `i_z`), que, juntas, respondem por quase 40% da importância e são essenciais para o "ajuste fino" da classificação, especialmente na diferenciação entre GALAXY e QSO. O painel de visualização Cor vs. Redshift (Figura 3) ilustra esta dinâmica, mostrando como as diferentes cores contribuem para separar as nuvens de GALAXY e QSO em regiões distintas do espaço de *features*, mesmo com a sobreposição existente.

Nossos resultados, com o Random Forest alcançando 97.84% de acurácia no SDSS DR17, são consistentes e altamente competitivos com os estudos mais recentes na área. Por exemplo, Clarke et al. (2020), ao classificar objetos do SDSS utilizando uma abordagem de Machine Learning incluindo Random Forest, reportaram acurácias na faixa de ~97-98% para seus melhores modelos em tarefas similares. De forma similar, Beck et al. (2022), focando na classificação fotométrica no SDSS, também demonstraram a alta performance de algoritmos

baseados em árvore. Adicionalmente, Kumar et al. (2021), em um estudo comparativo utilizando dados do SDSS DR14 (10.000 amostras), identificaram o Random Forest como o algoritmo superior entre os testados (SVM, Naive Bayes, k-NN), alcançando uma acurácia notável de 98.66%. A performance robusta do Random Forest em nosso estudo (97.84% em 100.000 amostras do DR17), embora ligeiramente inferior à de Kumar et al. (2021), reforça sua posição como uma ferramenta de primeira linha para esta tarefa. As pequenas diferenças de performance podem ser atribuídas a variações nas versões dos dados (DR14 vs DR17), ao tamanho do dataset, às features exatas utilizadas ou a etapas de otimização de hiperparâmetros não realizadas neste trabalho.

Embora o Random Forest tenha apresentado o melhor desempenho, reconhecemos como limitação deste estudo a não realização de uma otimização sistemática de hiperparâmetros para os modelos testados. Tal otimização poderia, potencialmente, elevar ainda mais a performance, embora às custas de maior complexidade computacional.

Em suma, a combinação de engenharia de features (criação de cores) com um modelo de ensemble robusto como o Random Forest provou ser uma estratégia altamente eficaz para a classificação de objetos no SDSS DR17, fornecendo uma ferramenta valiosa para a análise automatizada de grandes volumes de dados astronômicos.

6. CONSIDERAÇÕES FINAIS

Este trabalho investigou a aplicação e a performance comparativa de algoritmos de *Machine Learning* para a classificação automática de objetos celestes (Estrela, Galáxia e Quasar) utilizando dados da 17ª liberação do *Sloan Digital Sky Survey* (SDSS DR17). Através de uma metodologia que incluiu pré-processamento, engenharia de *features* com a criação de "cores" astronômicas, e a avaliação sequencial de modelos de complexidade crescente — Regressão Logística, Árvore de Decisão e *Random Forest* — demonstramos a eficácia destas técnicas para lidar com a classificação em larga escala.

Confirmamos a importância crucial do redshift como preditor primário, especialmente para a identificação de estrelas, e avaliamos o papel complementar das cores astronômicas (r_i , g_r , u_g , i_z) no refinamento da classificação, particularmente na complexa fronteira entre Galáxias e Quasares. A análise comparativa dos modelos estabeleceu o *Random Forest* como a abordagem superior, alcançando uma acurácia total de 97.84% e o melhor desempenho na classificação da classe Quasar (recall de 0.93). Estes resultados estão

alinhados e são competitivos com os reportados em estudos recentes na literatura (Clarke et al., 2020; Beck et al., 2022; Kumar et al., 2021).

Concluimos que o *Random Forest*, combinado com uma engenharia de *features* apropriada, representa uma solução robusta, precisa e eficaz para a classificação automatizada de objetos em grandes *datasets* astronômicos estruturados como o SDSS. A metodologia aqui apresentada oferece uma ferramenta valiosa para a análise científica em larga escala, permitindo a exploração eficiente dos vastos arquivos de dados gerados pelos modernos levantamentos astronômicos.

Como trabalhos futuros, sugere-se a exploração de técnicas de otimização de hiperparâmetros para potencialmente refinar ainda mais a performance do modelo *Random Forest*, bem como a aplicação desta metodologia validada a *datasets* de outros levantamentos astronômicos ou a versões mais recentes dos dados do SDSS.

REFERÊNCIAS

- ABDURRO'UF, et al. The Seventeenth Data Release of the Sloan Digital Sky Surveys (SDSS DR17). **The Astrophysical Journal Supplement Series**, v. 259, n. 2, p. 35, mar. 2022. DOI: 10.3847/1538-4365/ac4414.
- ARLOT, S.; CELISSE, A. A survey of cross-validation procedures for model selection. **Statistics Surveys**, v. 4, p. 40–79, 2010.
- BECK, M., et al. Photometric classification of SDSS objects using machine learning. **Astronomy & Astrophysics**, v. 660, A64, abr. 2022. DOI: 10.1051/0004-6361/202142490.
- CLARKE, A. O., et al. Machine learning classification of SDSS survey objects. **Monthly Notices of the Royal Astronomical Society**, v. 497, n. 3, p. 3291–3301, set. 2020. DOI: 10.1093/mnras/staa2166.
- CORY-WRIGHT, R.; GÓMEZ, A. Optimal Cross-Validation for Sparse Linear Regression. **INFORMS Journal on Computing**, 2025.

FEDESORIANO. **Stellar Classification Dataset - SDSS17**. Kaggle, 2022. Disponível em: <https://www.kaggle.com/datasets/fedesoriano/stellar-classification-dataset-sdss17>. Acesso em: 27 out. 2025.

FLUKE, C. J.; JACOBS, C. Surveying the reach and impact of machine learning in astronomy. **WIREs Data Mining and Knowledge Discovery**, v. 10, n. 4, e1339, jul./ago. 2020. DOI: 10.1002/widm.1339.

GEISSER, S. The predictive sample reuse method with applications. **Journal of the American Statistical Association**, v. 70, n. 350, p. 320–328, 1975.

GRANDINI, M.; BAGLI, E.; VISANI, G. Metrics for multi-class classification: an overview. **arXiv preprint arXiv:2008.05756**, 2020.

HUNTER, J. D. Matplotlib: A 2D Graphics Environment. **Computing in Science & Engineering**, v. 9, n. 3, p. 90-95, mai./jun. 2007. DOI: 10.1109/MCSE.2007.55.

KOHAVI, R. A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. In: INTERNATIONAL JOINT CONFERENCE ON ARTIFICIAL INTELLIGENCE (IJCAI), 1995.

KUMAR, A.; GUPTA, S.; KHANNA, D. Comparative Analysis of Machine Learning Algorithms for Classification of Astronomical Objects. **Statistical Journal of the IAOS**, v. 37, n. 4, p. 1387-1403, dez. 2021. DOI: 10.3233/SJI-210850.

LEI, J. A Modern Theory of Cross-Validation through the Lens of Stability. **arXiv preprint arXiv:2505.23592v3**, 2025.

MCKINNEY, W. Data Structures for Statistical Computing in Python. In: **PROCEEDINGS OF THE 9TH PYTHON IN SCIENCE CONFERENCE (SCIPY 2010)**, Austin, TX, 2010. p. 51-56.

PEDREGOSA, F., et al. Scikit-learn: Machine Learning in Python. **Journal of Machine Learning Research**, v. 12, p. 2825-2830, out. 2011.

SANTOS, E. S. et al. Performance Variability of Machine Learning Models using Limited Data for Collusion Detection: A Case Study of the Brazilian Car Wash Operation. **Proceedings of the 39th Brazilian Symposium on Data Bases**, 2024.

STONE, M. Cross-validators choice and assessment of statistical predictions. **Journal of the Royal Statistical Society: Series B (Methodological)**, v. 36, n. 2, p. 111–147, 1974.

WASKOM, M. L. Seaborn: statistical data visualization. **Journal of Open Source Software**, v. 6, n. 60, p. 3021, abr. 2021. DOI: 10.21105/joss.03021.