UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE FÍSICA
CURSO DE BACHARELADO EM FÍSICA

Busca de galáxias de baixo brilho superficial por similaridade em grandes catálogos astronômicos

Autor: Marcos Tidball

Orientadora: Profa. Dra. Cristina Furlanetto

Porto Alegre, 19 de outubro de 2022

Sumário

Agradecimentos

Gostaria de agradecer com muito carinho a minha mãe, Leonice, e também os *missis*: Nimi, Pepi e a falecida Tofi, bem como *a casinha* por sempre me apoiarem a continuar me esforçando na graduação (e em outras ocasiões!) mesmo quando eu menos tinha vontade.

Obrigado também a meu pai, Carlos, por todo o apoio ao longo dos anos e também por ter expandido minha visão de mundo graças a viagens.

Finalmente, um grande agradecimento às minhas duas avós: Loni e Myrna, por terem me ensinado muito e influenciado muito o modo como vejo as coisas.

Obrigado à Milena por participar de uma boa parte dessa jornada da minha graduação e estar lá por mim.

Obrigado aos *fisicamigos* (ou *amiguísicos*) pelas inúmeras partidas de *truco* (e derivados) nos momentos onde mais devíamos estar estudando. Gostaria de enfatizar Mateus e Thiago pela parceria e pelo carregamento também!

Gostaria de agradecer à professora Cristina Furlanetto por ter me dado a oportunidade de realizar uma bolsa de iniciação científica no campo de inteligência artificial aplicado em astronomia. Similarmente, obrigado também ao CNPq e a FAPERGS por proporcionarem as bolsas para essa pesquisa. Sem essa oportunidade eu não sei onde eu estaria na minha vida profissional!

Também um agradecimento especial ao meu gestor no banco BTG Pactual, João Mota, por ter confiado a mim não só um emprego na área de ciência de dados (onde aprendo muito) como também acesso a recursos computacionais para que eu pudesse realizar essa pesquisa.

Por fim, um obrigado geral para todos que me apoiaram nessa jornada, é isso aí!

Resumo

Galáxias de baixo brilho superficial (LSBGs, do inglês *Low Surface Brightness Galaxies*), constituem um segmento importante da população de galáxias, porém, por serem objetos de baixo brilho, sua busca é difícil. A sua detecção geralmente é realizada utilizando uma combinação de métodos paramétricos e inspeção visual, o que se torna inviável para futuros levantamentos fotométricos que coletarão *terabytes* de dados por noite. Assim, nesse trabalho exploramos a utilização do algoritmo *Locality-Sensitive Hashing* para a realização de uma busca por similaridade aproximada de LSBGs em grandes catálogos astronômicos. Utilizamos 11670190 objetos do catálogo *DES Y3 Gold coadd* para criar um modelo de busca de similaridade baseado nas propriedades físicas dos objetos, desenvolvendo uma ferramenta capaz de encontrar novos candidatos a LSBGs com o uso de somente uma LSBG conhecida que faz parte desse catálogo. A partir de uma galáxia conhecida de uma amostra, conseguimos encontrar várias outras galáxias da mesma amostra, além de novas galáxias visualmente similares mas ainda não catalogadas. Apresentamos também o resultado de nosso modelo para a busca de artefatos, demonstrando a sua generalidade e o pouco tempo de busca necessário para retornar objetos similares. O código utilizado nesse trabalho está disponível em `https://github.com/zysymu/lsh-astro`.

Abstract

Low Surface Brightness Galaxies (LSBGs) constitute an important segment of the galaxy population, however, due to their low brightness, their search is challenging. The detection of LSBGs is usually done with a combination of parametric methods and visual inspection, which becomes unpractical for future astronomical surveys that will collect *petabytes* of data. Thus, in this work we explore the usage of Locality-Sensitive Hashing for the approximate similarity search of LSBGs in large astronomical catalogs. We use 11670190 objects from the DES Y3 Gold coadd catalog to create an approximate k-nearest neighbor model based on the physical properties of the objects, developing a tool able to find new LSBG candidates while using only one known LSBG. From just one labeled example we are able to find various already known LSBGs and many objects visually similar to LSBGs but not yet catalogued. Also, due to the generality of similarity search models, we are able to search for and recover other rare astronomical objects without the need of retraining or generating a large sample. Our code is available at `https://github.com/zysymu/lsh-astro`.

Lista de Figuras

Lista de Abreviaturas e Siglas

kNN	*k-Nearest Neighbors*. 6, 10–13, 15, 30
DECam	*Dark Energy Camera*. 19
DES	*Dark Energy Survey*. 19–21, 29
LSBGs	*Low Surface Brightness Galaxies*. 1–6, 19, 21, 24–26, 29
LSH	*Locality Sensitive Hashing*. 13, 15, 19, 21–24, 29, 30
OHE	*One-Hot Encoding*. 22
UDGs	*Ultra-Diffuse Galaxies*. 1–4

1 Introdução

Galáxias de baixo brilho superficial, do inglês *Low Surface Brightness Galaxies* (LSBGs), constituem um segmento importante e peculiar da população de galáxias no Universo. Caracterizadas por apresentaram baixo brilho superficial e serem difusas, podem ser facilmente neglicenciadas por causa de vieses observacionais. Mesmo assim, são uma classe de objetos importante para o entendimento do Universo. Acredita-se que uma parte significativa da matéria no Universo Local pode estar presente em fontes difusas, uma ideia suportada pela descoberta recente de galáxias ultra-difusas, do inglês *Ultra-Diffuse Galaxies* (UDGs), em aglomerados de galáxias e em ambientes isolados (Prole et al., 2019). Podemos observar um exemplo de galáxia ultra-difusa na Figura 1.

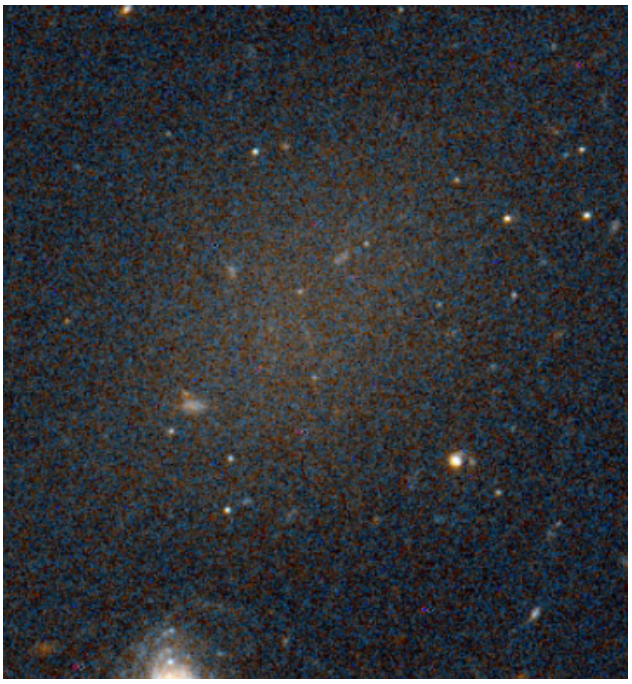


Figura 1: Exemplo de uma UDG.

Fonte: Dokkum et al. (2015)

Para entendermos melhor quais são as características que eles apresentam desses objetos e os mecanismos responsáveis pela sua formação e evolução, é necessário termos uma grande amostra completa de LSBGs com medidas físicas precisas. A detecção de LSBGs, porém, é uma tarefa difícil, dado que a maioria dos métodos de detecção de objetos astronômicos tem como foco fontes brilhantes. Os métodos mais tradicionais que resolvem esse problema envolvem muito tempo computacional e/ou requerem inspeção visual de grande quantidade de imagens, dificultando uma detecção automática em larga escala.

Nota-se que as novas gerações de levantamentos astronômicos, como *Euclid*[1] e *LSST*[2] trarão quantidades cada vez maiores de dados para serem analisados, com o último, por exemplo, esperado a produzir 20 TB de dados por noite, observando aproximadamente 10 bilhões de galáxias em seus 10 anos de atividade. Assim, faz-se necessário o desenvolvimento de técnicas computacionais que permitam o processamento de quantidades enormes de dados para encontrar objetos astronômicos de um modo eficiente que sejam de interesse para diferentes grupos de pesquisadores.

O objetivo deste trabalho é, assim, propor uma possível solução geral para esse problema, utilizando LSBGs como nosso foco de estudo.

A seguir, será feita uma revisão sobre as galáxias de baixo brilho superficial e sobre os métodos de detecção já existentes que permitem a sua identificação.

1.1 Galáxias de baixo brilho superficial

As LSBGs são caracterizadas, como seu nome implica, pelo seu baixo brilho superficial, sendo convencionalmente definidas como galáxias que apresentam brilho superficial central mais fraco que o céu noturno (Bothun et al., 1997). Rigorosamente, as LSBGs são definidas como galáxias que apresentam brilho superficial central $\mu_0(B) \geq 23.0$ mag arcsec^{-2} (Yi et al., 2022). Sua existência foi inicialmente proposta por Disney (1976), mas devido ao seu baixo brilho, só foram observadas de fato no final da década de 1980 (Bothun et al., 1987). Recentemente, telescópios com tecnologias inovadoras e especializados em características de baixo brilho foram capazes de identificar LSBGs novas e reacenderam o interesse nesse tipo de objeto (Dokkum et al., 2015; Lim et al., 2018).

Acredita-se que essas galáxias contribuem pouco para luminosidade, mas que podem representar $\sim 15\%$ da massa do Universo atual (Driver, 1999). Além disso, elas apresentam-se em diferentes tamanhos e em diferentes ambientes, como satélites ultra-fracos da Via Láctea (Simon, 2019), satélites de outras galáxias próximas (Carlsten et al., 2021) e membros de aglomerados massivos de galáxias (Dokkum et al., 2015). Entre as principais perguntas sobre as LSBGs estão: como elas acabaram presentes em tantos ambientes diferentes? Como elas se formaram e evoluíram? O ambiente tem impacto na sua formação e evolução? Como galáxias com tão baixa densidade de estrelas se mantêm coesas? Seriam elas galáxias anômalas? As respostas para essas questões podem trazer informações importantes sobre modelos cosmológicos e de evolução de galáxias.

Um tipo especial de galáxia de baixo brilho, as UDGs, detectadas inicialmente por Dokkum et al. (2015), despertaram muito interesse devido, por exemplo, às diferentes questões ainda muito debatidas sobre suas propriedades (Sales et al., 2020). Essas galáxias apresentam um raio efetivo de $1.5 \leq R_e \leq 4.6$ kpc e brilho superficial central de $24.0 \leq \mu_0(B) \leq 26.0$ mag arcsec^{-2}. Sendo assim, seu tamanho é similar à Via Láctea, mas seu brilho superficial é consideravelmente inferior, sendo as galáxias mais extremas da população de LSBGs (Conselice, 2018). Podemos

[1] https://www.euclid-ec.org/
[2] https://www.lsst.org/

observar a relação tamanho-luminosidade de LSBGs e UDGs com relação a outros sistemas de estrelas na Figura 2.

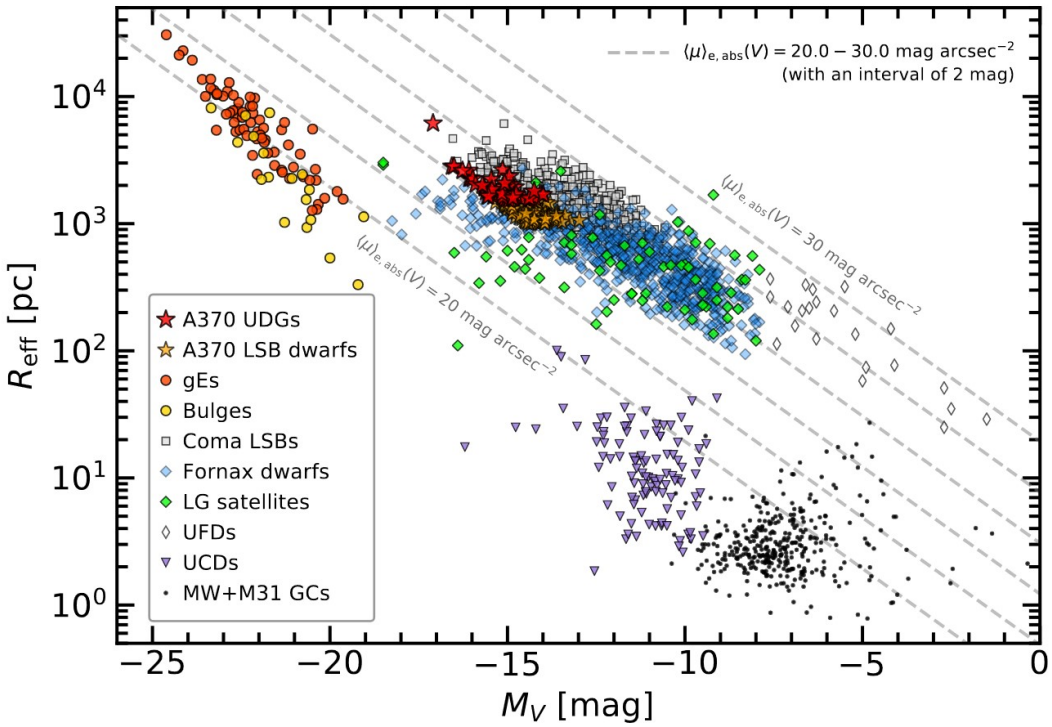


Figura 2: A relação tamanho-luminosidade de LSBGs (símbolos de estrelas amarelas) e UDGs (símbolos de estrelas vermelhas) em comparação com outros sistemas de estrelas. No eixo vertical temos o raio efetivo medido e no eixo horizontal temos a magnitude dos objetos.

Fonte: Lee et al. (2020)

Considerando as propriedades de massa estelar e metalicidade, LSBGs, enquanto exibem uma larga distribuição de tamanhos físicos e massas do halo, são muito similares a galáxias anãs (Prole et al., 2019). Isso em conjunto com o fato que algumas observações indicam que UDGs podem não possuir matéria escura (van Dokkum et al., 2018) sugere a existência de diversos mecanismos distintos de formação e evolução de galáxias ultra-difusas. Há hipóteses que elas podem se formar de modo secular tanto em aglomerados quanto isolados em campo (Amorisco et al., 2018) por meio de uma expulsão de gás que gera uma expansão de matéria escura e estelar, produzindo UDGs que possuem tamanhos grandes mesmo com massas de halo similares a de galáxias anãs (Di Cintio et al., 2017). Propõe-se, também, UDGs se formam a partir de galáxias do tamanho da Via Láctea que, por alguma razão, formam menos estrelas, fazendo com que sejam difusas. Isso explicaria o grande tamanho das galáxias e a menor quantidade de estrelas delas em comparação com outras galáxias da mesma massa (van Dokkum et al., 2015). Outra teoria para sua formação considera que UDGs podem ter sido criadas a partir de um processo de transformação de galáxias como consequência de forças de maré, gerando remanescentes de baixo brilho superficial (Sales et al., 2020).

Outra questão importante relacionada às LSBGs é o fato de que suas propriedades variam conforme o ambiente onde elas se encontram: LSBGs que habitam aglomerados de galáxias são, em sua maioria, de sequência vermelha e com alta relação massa-luminosidade. Enquanto isso, LSBGs que existem em ambientes de baixa densidade são, em sua maioria, anãs (Prole et al., 2019). Nota-se também que grande quantidade das UDGs localizam-se em aglomerados de galáxias, um fato incomum para galáxias de massa e densidade estelar baixas (Lim et al., 2018), porém essas galáxias estão presentes majoritariamente fora do centro do aglomerado, indicando que elas não conseguem se formar de modo muito eficiente em regiões de alta densidade de galáxias (Dokkum et al., 2015). Nota-se também que LSBGs e UDGs podem possuir ricos sistemas de aglomerados globulares, os quais podem ser usados para estimativas de massa (Beasley et al., 2016).

O modelo padrão cosmológico (ΛCDM) assume que o processo de formação de galáxias é feito de modo hierárquico, com galáxias pequenas se agregando para formar galáxias maiores. LSBGs, com seus valores extremos de tamanho-luminosidade, são objetos de interesse para testar esses modelos e checar o quão completos eles são (Sales et al., 2020). Galáxias de baixo brilho superficial constituem uma fração significativa da população de galáxias, apresentando um papel importante no estudo da formação e evolução de galáxias no Universo (Impey and Bothun, 1997). Assim, é de suma importância conhecer e estudar de modo detalhado LSBGs e como elas e suas propriedades se distribuem para entender melhor a natureza dessas galáxias.

1.2 Detecção automática de LSBGs

Os métodos utilizados para a detecção de LSBGs e UDGs em geral dividem-se em duas categorias: métodos paramétricos e métodos baseados em aprendizado de máquina. Os métodos paramétricos fazem uso de modelos que estimam parâmetros fotométricos dos objetos, os quais são então utilizados para a realização de uma seleção da amostra, normalmente seguidos de uma inspeção visual. O trabalho de Zaritsky et al. (2018) utilizou diversos algoritmos para subtrair fontes brilhantes de imagens, aplicar filtros sobre elas e realizar cortes baseados em valores obtidos a partir de modelos paramétricos. Prole et al. (2019), por sua vez, utilizando a ferramenta *MTObjects* (Teeninga et al., 2015) detectaram fontes e criaram mapas de segmentação que permitiram a eliminação de objetos com parâmetros não característicos de UDGs, como brilho superficial e raio efetivo elevados. Também nota-se o trabalho de Li et al. (2022) que emprega o fato de LSBGs e UDGs possuírem ricos sistemas de aglomerados globulares para procurar por essas galáxias através de sobredensidade de fontes pontuais em imagens astronômicas

Técnicas paramétricas para a detecção de objetos, enquanto mais precisas, necessitam de muito tempo de processamento computacional e geralmente muita análise manual. O processo de estimativa de parâmetros necessita de valores inicias que, caso muito diferentes dos valores corretos podem fazer com que o modelo fique preso em algum mínimo local, estimando parâmetros que não fazem sentido físico (Pearson et al., 2021). Por essa razão, enquanto esses métodos são de suma importância para uma análise mais profunda de itens

detectados, devido a sua forte dependência de valores iniciais e alta exigência computacional, eles se tornam inviáveis para grandes quantidades de dados.

Por outro lado, métodos de inteligência artificial que fazem uso de aprendizado de máquina permitem que o usuário não precise ficar ajustando parâmetros para cada tipo de objeto, sendo necessário somente uma amostra de treinamento grande o suficiente e completa para a obtenção de resultados interessantes (Hastie et al., 2001). Essa tecnologia já foi utilizada em diversas áreas da astronomia com a intenção de detectar objetos específicos de um modo eficiente e preciso (Metcalf et al., 2019; Bom et al., 2022; Valenzuela and Pichara, 2018; Alexander et al., 2021).

Tratando-se de LSBGs, o trabalho de Tanoglidis et al. (2021a) utilizou aprendizado de máquina (mais especificamente, *Support Vector Machines*) para treinar um modelo em dados tabulares de propriedades físicas — magnitude, brilho superficial, elipticidade, dimensões e fluxo — para realizar automaticamente um corte de $\sim$ 420000 objetos para $\sim$ 44000 objetos que foram classificados como LSBG pelo modelo. Após inspeção visual dos objetos classificados como LSBGs e uma estimativa de parâmetros por meio de métodos paramétricos para um corte final baseado no índice de Sérsic e correção por extinção, chegou-se em uma amostra de $\sim$ 23790 LSBGs. Alguns exemplos dessa amostra podem ser observadas na Figura 11. Nota-se pelos números, assim, o poder do aprendizado de máquina de reduzir grandes amostras de objetos gerais para amostras menores que contêm somente objetos similares e de interesse aos pesquisadores.

Contudo, esse trabalho fez uso de técnicas de aprendizado supervisionado, onde é necessária uma amostra grande de dados positivos (nesse caso, LSBGs) e dados negativos (nesse caso, objetos que não sejam LSBGs) para o treinamento do modelo. Para criar essas amostras Tanoglidis et al. (2021a) inspecionaram $\sim$ 6000 objetos manualmente. Uma alternativa, utilizada por Tanoglidis et al. (2021b) envolve o uso de redes neurais convolucionais, uma técnica que permite a inferência do tipo de objeto a partir dos dados de imagem. Porém, para esse modelo, também é necessário uma amostra que contenha exemplos positivos e negativos para o treinamento. Redes neurais convolucionais geralmente apresentam resultados mais precisos que modelos que utilizam dados estruturados (Metcalf et al., 2019) em problemas de classificação, porém sua exigência computacional é muito maior, tornando-se de difícil aplicação para grandes catálogos. Além disso, como necessitam de amostras de treinamento, só serão capazes de classificar objetos nas classes presentes na amostra, não sendo um modelo geral. Será que podemos obter resultados semelhantes mas com uma amostra consideravelmente menor de dados?

1.3 Busca de objetos astronômicos por similaridade

O problema de busca por similaridade, também conhecido como *busca de vizinhos próximos*, é definido pela busca dos itens mais similares (mais próximos) de um dado chave (um exemplo do tipo de objeto que queremos encontrar) (Wang et al., 2014). Esse método é muito utilizado por ferramentas de busca para encontrar imagens, textos e outros dados semelhantes a um

objeto específico, chamado de chave (Bhatia and Author, 2010). No contexto de objetos astronômicos, essas técnicas permitem a descoberta de corpos celestes similares a uma chave.

Stein et al. (2022) utilizou busca de similaridade para fazer um sistema que detecta lentes gravitacionais. Nesse trabalho, os autores utilizam aprendizado auto-supervisionado para encontrar modos de representar imagens de objetos astronômicos como vetores, diminuindo sua dimensionalidade mas ainda retendo informações relevantes. A partir disso, foi realizada uma busca de similaridade sobre os vetores das imagens. O interessante, e diferente de outros métodos de aprendizado de máquina comumente utilizados para classificação de objetos astronômicos, é que não foi necessária a criação de uma amostra de lentes gravitacionais para o treinamento do modelo. Efetivamente, isso faz com que o modelo seja agnóstico a um tipo de objetos. Enquanto o classificador de Tanoglidis et al. (2021a) só conseguia fazer a distinção entre LSBGs e não-LSBGs, a busca por similaridade encontra outros objetos com a mesma natureza da chave: se a chave for uma lente gravitacional, são encontradas lentes gravitacionais similares; se a chave for uma galáxia de baixo brilho superficial, são encontradas galáxias de baixo brilho superficial similares.

O poder desses métodos, segundo Hastie et al. (2001), se baseia no fato de que eles são livres de modelos, não sendo necessário o treinamento de diferentes parâmetros a partir de uma amostra de objetos positivos (por exemplo, LSBGs) e negativos (por exemplo, não-LSBGs). Devido a sua natureza altamente não estruturada, geralmente não são muito úteis para um entendimento melhor da natureza da relação entre características de diferentes itens que são classificados como próximos. Porém, como técnicas preditivas de *caixa preta* (modelos que produzem informações úteis sem revelar informações sobre sua tomada de decisão interna), são os métodos de aprendizado de máquina que geralmente atingem a melhor performance em problemas que usam dados reais, mesmo sem necessitar de uma amostra rotulada.

Técnicas de busca de similaridade, mesmo apresentando alto poder para encontrar objetos de uma mesma classe, têm um uso relativamente baixo em astronomia em comparação com outros modelos de aprendizado de máquina. Isso se deve provavelmente ao fato de que é necessário grande poder computacional para que as buscas sejam executadas em grandes quantidades de dados, como as presentes nos atuais e futuros levantamentos astronômicos. Nota-se porém trabalhos como o de Li et al. (2008), que utilizaram o algoritmo k-vizinhos próximos, do inglês *k-Nearest Neighbors* (kNN), para classificar objetos celestiais do levantamento *ROSAT All-Sky survey*. Foram utilizadas propriedades físicas dos objetos para diferenciar núcleos galácticos ativos de estrelas e galáxias normais, com os autores enaltecendo a alta performance preditiva e explicabilidade do método.

O método kNN, enquanto a base para os algoritmos de busca de vizinhos próximos, é um método muito ineficiente para grandes quantidades de dados, como veremos na seção 2.2. Devido a isso, alguns trabalhos dedicam-se ao estudo de técnicas que aumentam a eficiência do método para contextos astronômicos. Isso é feito por meio de otimizações do algoritmo e técnicas de paralelização para otimizar a busca por objetos similares (Heinermann et al., 2013; Łukasik et al., 2019).

1.4 Objetivos e estrutura

O objetivo desse trabalho é, assim, criar um método de busca de similaridade para objetos astronômicos a partir de propriedades físicas usando dados tabulares. A intenção é que isso permita o encontro de objetos fisicamente similares de modo eficiente e automático. Utilizaremos galáxias de baixo brilho superficial como o foco do nosso trabalho e para testar o nosso modelo. Contudo, devido a alta generalização desse modelo, ele também poderá ser utilizado para encontrar outros tipos de objetos astronômicos automaticamente.

O trabalho se estrutura da seguinte forma: na seção 2.1 apresentamos o modelo de busca de vizinhos próximos, que é a base do modelo que utilizaremos; na seção 2.2 falamos sobre as limitações do modelo de busca de vizinhos próximos; na seção 2.3 apresentamos uma solução para as limitações da busca de vizinhos próximos, explicando o modelo que utilizaremos. Na seção 3.1 falaremos sobre os dados que utilizaremos; na seção 3.2 explicaremos e justificaremos o modo como processamos e preparamos os dados para serem utilizados no modelo; na seção 3.3 mostraremos o modo como treinamos o modelo e os parâmetros que utilizamos; na seção 3.4 mostraremos os resultados do nosso modelo aplicado nos nossos dados. Finalmente, no capítulo 4 apresentaremos os próximos passos para o trabalho, assim como possíveis aplicações do mesmo.

2 *Locality Sensitive Hashing* (LSH)

Neste capítulo apresentaremos os fundamentos teóricos do algoritmo de busca de vizinhos próximos, que serve como base para a busca de similaridade. Descrevemos também suas ineficiências de um ponto de vista de complexidade temporal computacional, que fazem com que o algoritmo se torne inviável para buscas com grandes números de objetos que apresentam muitos parâmetros. Por fim, explicamos o funcionamento do algoritmo que utilizamos em para nossa aplicação: *Locality Sensitive Hashing*, uma maneira altamente eficiente de realizar uma busca de similaridade aproximada que apresenta uma ótima performance.

2.1 Busca de vizinhos próximos

A busca de vizinhos próximos pode ser descrita como um problema de busca de distância, onde é possível quantificar o quão similares objetos guardados numa base de dados são a um objeto chave (Zezula et al., 2006). Para realizar as buscas de itens similares utilizamos o conceito de *espaços métricos* (Kelley, 1955), que permite escrever o problema de busca como: seja $\mathcal{D}$ um domínio, d uma medida de distância em $\mathcal{D}$, e $(\mathcal{D}, d)$ um espaço métrico, dado um conjunto $X \subseteq \mathcal{D}$ de n elementos, estruture os dados de modo que as consultas de proximidade são respondidas eficientemente.

De um ponto de vista prático, X pode ser um conjunto de dados de itens que têm valores do domínio $\mathcal{D}$, com d sendo uma função distância definida para um par arbitrário de objetos de $\mathcal{D}$.

Em um espaço métrico, a única operação possível em objetos é o cálculo da função distância d em pares de objetos de modo a satisfazer a *desigualdade triangular*, que é definida como:

Teorema. $\forall\ x, y, z \in \mathcal{D},\ d(x, z) \leq d(x, y) + d(y, z)$

No nosso caso, consideramos um *espaço de coordenadas*[3] — um caso especial do espaço métrico — onde os objetos em questão são vetores. Assim, além de computar a distância entre pontos, temos que cada vetor está unicamente localizado no espaço. Além disso, novos vetores podem ser construídos a partir de operações matemáticas nos vetores já existentes.

As funções de distância d de espaços métricos definem um modo de quantificar a proximidade de objetos em um domínio. Para nosso estudo, iremos considerar a *distância Euclidiana* devido ao seu amplo estudo e forte aplicabilidade para dados de valores contínuos (Wang et al., 2014). Essa distância é um caso especial da família de funções de *distância de Minkowski*, designadas como as métricas L_p, que são definidas em vetores m-dimensionais de números reais, segundo Zezula et al. (2006), como:

[3]No nosso caso, as coordenadas são valores das propriedades que cada objeto apresenta, não sendo relacionadas com a posição astronômica do objeto no céu.

$$L_p\left[(a_1, a_2, ..., a_m), (b_1, b_2, ..., b_m)\right] = \sqrt[p]{\sum_{i=1}^{m} |a_i - b_i|^p} \tag{1}$$

Na equação 1, $(a_1, a_2, ..., a_m)$ e $(b_1, b_2, ..., b_m)$ representam os parâmetros dos dados, que podem ser coordenadas ou, no nosso caso, medidas físicas. No caso de um espaço de coordenadas bidimensional ($m = 2$), por exemplo, teríamos (a_1, a_2) e (b_1, b_2), com a_1 e b_1 sendo as coordenadas horizontais e a_2 e b_2 sendo as coordenadas verticais. Diferentes valores de p geram diferentes métricas, com a distância Euclidiana sendo a métrica gerada por $p = 2$, assim:

$$L_2 = \left[(a_1, a_2, ..., a_m), (b_1, b_2, ..., b_m)\right] = \sqrt{\sum_{i=1}^{m} |a_i - b_i|^2}\ , \tag{2}$$

que define a métrica que usaremos para nosso trabalho, assim fixamos $d = L_2$. A Figura 3 mostra alguns membros da família L_p, com as formas demonstrando os pontos de um espaço vetorial bidimensional que estão na mesma distância do ponto central.

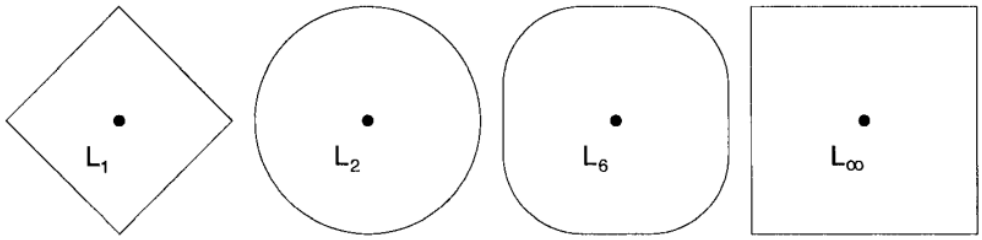


Figura 3: O conjunto de pontos a uma distâncias constante do ponto central para diferentes funções de distância L_p.

Fonte: Zezula et al. (2006)

Com o espaço sobre o qual trabalhamos e a função de distância que usaremos, resta definir a chamada de similaridade que será empregada. Um pedido de similaridade é definido a partir de um objeto chave q e uma restrição na extensão da proximidade requerida (até qual distância de q realizamos a busca). A resposta da chamada retorna todos os objetos que satisfazem as condições de seleção, formando os objetos mais próximos da chave q.

A busca de vizinhos próximos é um dos possíveis tipos de chamada que encontra os objetos mais próximos de q (Zezula et al., 2006). No caso, a restrição utilizada é o número de vizinhos mais próximos k. Esse é o conceito base da busca kNN, que retorna os k vizinhos mais próximos de algum objeto q.

Formalizando esse conceito, definimos o alcance $R(q, r)$ como a função que retorna todos objetos que estão a uma distância r de q:

$$R(q, r) = \{x \in X,\ d(q, x) \leq r\} \tag{3}$$

Nota-se que o objeto chave q não precisa ser um elemento do conjunto $X \subseteq \mathcal{D}$, mas é necessário que $q \subseteq \mathcal{D}$. De um ponto de vista prático, isso significa que enquanto o item de busca não precisa participar do nosso conjunto de dados, mas ele precisa ter o mesmo formato dos itens presentes no mesmo. A Figura 4 demonstra uma busca por alcance.

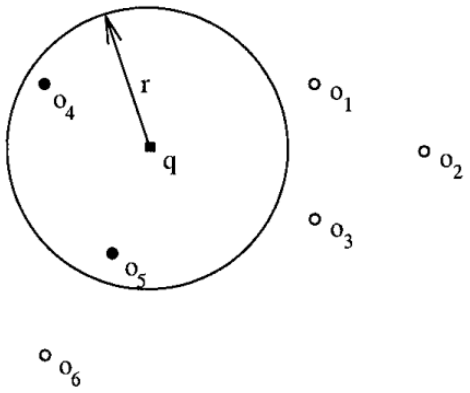


Figura 4: Um exemplo de uma busca por alcance usando a função $R(q, r)$ num espaço de coordenadas bidimensional com objetos $o_1, o_2, ..., o_6$

Fonte: Zezula et al. (2006)

Assim, podemos definir matematicamente a busca kNN como:

$$\text{kNN}(q, k) = \{R \subseteq X, |R| = k \wedge \forall\ x \in R,\ y \in X - R :\ d(q, x) \leq d(q, y)\} \tag{4}$$

Em casos onde mais de um objeto está na mesma distância da chave q, os empates são resolvidos escolhendo a ordem que os objetos de mesma distância são ordenados de forma aleatória. A Figura 5 ilustra uma busca que utiliza a Equação 4 com $k = 3$.

Com os fundamentos matemáticos definidos, podemos escrever o algoritmo computacional usado para realizar a busca de vizinhos próximos. A maneira mais simples de realizar a busca num conjunto de dados envolve comparar o objeto q com todos os objetos do conjunto. Isso se deve ao fato de que q é um parâmetro que varia de busca para busca, de modo que não seja possível garantir o ordenamento dos dados com respeito a q.

Assim, a criação do conjunto de vizinhos mais próximos, também chamado de *conjunto resposta*, é um processo incremental. Tendo $k \leq n$, a versão inicial desse conjunto é formado pelos primeiro k objetos ordenados com respeito a sua distância de q. Para todos os outros elementos, um objeto o_i é inserido no conjunto de vizinhos mais próximos somente se $d(q, o_i) < d(q, o_k)$, onde o_k é o k-ésimo vizinho mais próximo de q no dado momento. Sempre que um novo objeto é inserido no conjunto resposta, o k-ésimo vizinho anterior é eliminado.

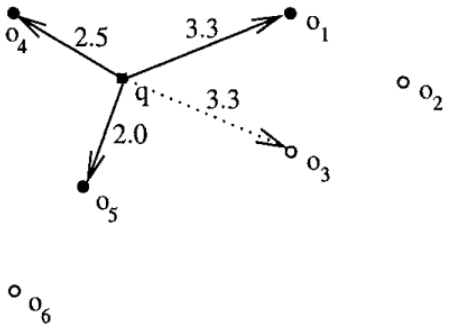


Figura 5: Um exemplo de uma busca de vizinhos próximos (KNN) usando a função kNN$(q, 3)$ num espaço de coordenadas bidimensional. Aqui, os objetos o_1 e o_3 estão na mesma distância do objetos chave, fazendo com que o objeto o_1 seja escolhido como o terceiro vizinho mais próximo de forma aleatória.

Fonte: Zezula et al. (2006)

2.2 Limitações da busca absoluta

Uma das desvantagens do kNN é o gasto computacional que ocorre tanto na busca de vizinhos quanto na necessidade de guardar o conjunto de todos os dados (Hastie et al., 2001). Existem algoritmos eficientes para a busca exata de vizinhos próximos em casos de dados de baixas dimensões (i.e. poucas colunas), mas esse problema se torna mais complexo em casos onde temos muitas dimensões (Wang et al., 2014).

Algoritmos de busca de vizinhos próximos que utilizam métricas de distância de Minkowski — (a família de funções definidas na Equação 1) — apresentam uma complexidade temporal no formato $\mathcal{O}(nm)$, onde n é o número de elementos e m é o número de medidas que descrevem cada elemento (Cunningham and Delany, 2020). Isso significa que o número de comparações necessárias aumenta linearmente com o número de dados que temos.

Assim, obtermos os k pontos mais próximos da nossa chave q, precisamos realizar o cálculo da distância d entre q e cada um dos n pontos do nosso conjunto de dados. Isso faz com que seja necessário iterar sobre todos os nossos dados para obter os vizinhos mais próximos. Para grandes quantidades de dados, isso se torna inviável computacionalmente. Somado a isso, o algoritmo também depende do tamanho dos nossos dados, o que significa que temos de realizar o cálculo de comparação de distância para cada uma das medidas de nossos dados, o que se torna inviável quando apresentamos dados de grandes dimensões.

Felizmente, para muitas aplicações, não é essencial que tenhamos os vizinhos próximos absolutos. Itens que estejam razoavelmente próximos já podem ser suficientes para permitirem descobertas de interesse (Cunningham and Delany, 2020). Esta é a ideia principal de *métodos aproximados de kNN*

2.3 Busca aproximada de vizinhos próximos

Nesse trabalho usaremos uma das estratégias mais populares para o kNN aproximado em dados de grandes dimensões: *Locality Sensitive Hashing* (LSH). Segundo Leskovec et al. (2020), o funcionamento dessa família de técnicas se baseia no uso de funções *hash* para agrupar dados em *buckets*. Feito isso, no momento de inferência só é necessário ao buscar os vizinhos de q examinar os itens que estejam no mesmo *bucket* de q.

O principal conceito por trás desses algoritmos é a estrutura de dados de tabelas *hash*. Esse tipo de estrutura na prática apresenta uma performance muito boa, sendo necessário um tempo constante $\mathcal{O}(1)$ para encontrar um elemento na tabela *hash* (Cormen et al., 2009).

Uma das estruturas mais básicas para o armazenamento de dados é o vetor. Cada posição possível de um vetor apresenta um índice correspondente. Quando queremos um elemento x de um vetor porém, a não ser que tivermos a informação da posição onde x está armazenado, precisamos realizar (em casos onde o vetor não segue nenhuma ordem) uma busca linear. Essa busca se baseia na análise de cada posição do vetor até encontrarmos o elemento x. Um esquema da busca está presente na Figura 6, onde nota-se o tempo $O(|\vec{v}|)$, com $|\vec{v}|$ sendo o tamanho do vetor, necessário para encontrar x.

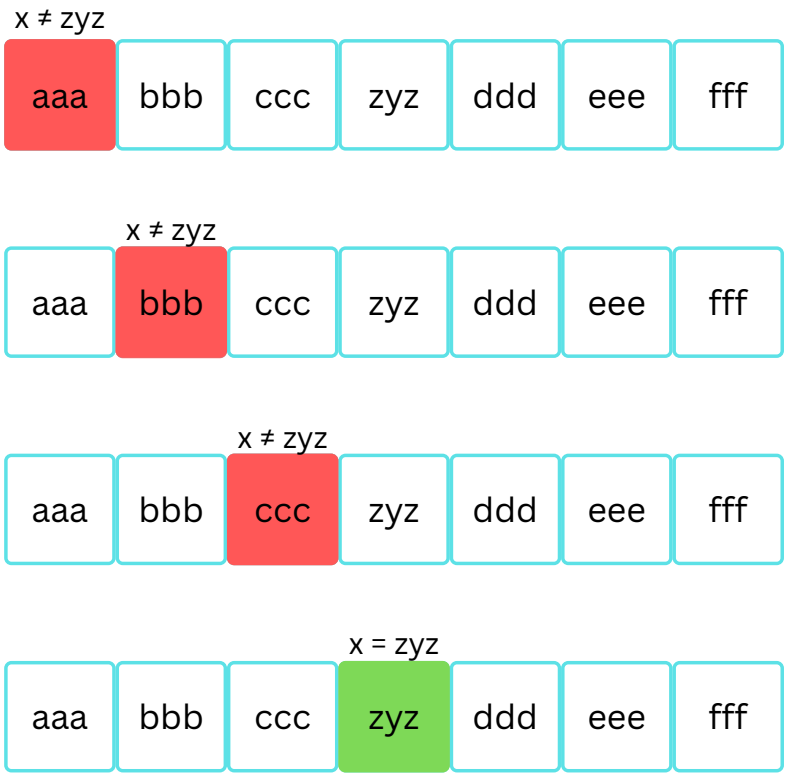


Figura 6: Exemplo de busca linear em um vetor onde queremos encontrar o elemento de valor $x = \texttt{zyz}$.

Tabelas *hash* têm como intenção deixar essa busca muito mais eficiente: ao invés de guardar um elemento no vetor numa posição arbitrária, podemos guardá-lo na posição $h(x)$. Ou seja, usamos uma função *hash* h para calcular a posição onde x será armazenado a partir do valor de x. Dizemos, assim, que o elemento x é *hashado* para o *bucket* $h(x)$ (Cormen et al., 2009). Assim, é possível encontrarmos um elemento específico em um vetor sem a necessidade de realizar uma busca, mas simplesmente desenvolvendo uma função x que mapeie elementos diferentes para posições diferentes, como podemos observar no exemplo da Figura 7.

Figura 7: Exemplo de uma tabela *hash* que utiliza uma função *hash* que encontra o *bucket* de elementos somando a posição das letras que compõem o elemento e realizando a operação de módulo (que retorna o resto da divisão) pelo tamanho da tabela.

Dependendo da definição de h, haverão casos onde elementos com valores diferentes podem ser *hashados* para a mesma posição na tabela. Há diversas técnicas que resolvem esse problema de modo eficiente, como o encadeamento de elementos usando listas encadeadas, mas esse tópico é além do escopo do trabalho (Cormen et al., 2009, descreve o funcionamento de alguns desses mecanismos em detalhe).

Nota-se que uma das maiores dificuldades em métodos de *hashing* é a definição de uma

função h. Uma função boa satisfaz (mesmo que aproximadamente) a hipótese de que cada elemento tem a mesma chance de mapear cada uma das posições da tabela. Isso deve ser feito independentemente de onde os outros elementos estejam posicionados (Cormen et al., 2009). Dificilmente conseguimos checar essa condição devido ao fato de raramente sabermos a distribuição de onde os elementos são tirados.

Voltando ao LSH, a intenção é obter alguma função *hash* tal que que itens similares ocuparão o mesmo *bucket* enquanto itens dissimilares ocuparão *buckets* diferentes, fazendo com que na busca somente uma fração dos itens tenha de ser examinada. Essa ideia permite aumentar bastante a eficiência computacional ao processar grandes conjuntos de dados de alta dimensionalidade. Porém, perdemos a precisão do kNN devido ao fato que itens similares podem não ser considerados por estarem em outro *bucket* (*falso negativos*) e itens dissimilares podem ser considerados vizinhos próximos por estarem no mesmo *bucket* (*falso positivos*).

O desafio dessa técnica é criar uma função *hash* que seja sensível à distância. Desse modo, podemos definir o problema de LSH rigorosamente como (Datar et al., 2004): Seja $\mathcal{D}$ um domínio, d uma medida de distância em $\mathcal{D}$ e $(\mathcal{D}, d)$ um espaço métrico. Considerando P como a probabilidade, encontre uma família de funções *hash* H tal que cada função *hash* $h \in H$ da família obedeça:

$$\forall\, x, y \in \mathcal{D},\ d(x,y) \leq r_1 \Rightarrow P(h(x) = h(y)) \geq p_1\ \wedge\ d(x,y) \geq r_2 \Rightarrow P(h(x) = h(y)) \leq p_2\ ,$$

onde r_1 e r_2 são distâncias e p_1 e p_2 são probabilidades. Para H ser computacionalmente eficiente é necessário que cada uma de suas funções satisfaça as desigualdades $p_1 > p_2$ e $r_1 < r_2$.

Assim, queremos encontrar uma função onde, se itens forem próximos, teremos uma alta probabilidade de que a função *hash* de cada um seja igual, fazendo com que eles fiquem no mesmo *bucket*. Além disso, também queremos que se itens forem distantes, teremos uma alta probabilidade de que a função *hash* de cada um seja diferente, fazendo com que eles fiquem em *buckets* diferentes.

Dessa definição de LSH, temos que os falso positivos são definidos como itens distantes ($d(x,y) \geq r_2$) *hashados* no mesmo *bucket* e que falso negativos são itens próximos ($d(x,y) \leq r_1$) colocados em *buckets* diferentes.

Diferentes medidas de similaridade d apresentam diferentes funções *hash* que conservam a distância dos dados. No caso de distância no formato da Equação 1, as funções *hash* agrupam os dados em vetores unidimensionais — que são efetivamente tabelas *hash* (Leskovec et al., 2020). Inicialmente, um número t de tabelas *hash* (que operam como vetores unidimensionais) é gerado com orientações aleatórias que cortam o espaço métrico. Esses vetores são então particionados em *buckets* de tamanho s. Os pontos do espaço métrico são projetados em cada um dos t vetores, sendo *hashados* no *bucket* que coincide com a coordenada de sua projeção (Wang et al., 2014).

Visualmente, podemos considerar um espaço bidimensional somente com uma tabela

hash (um vetor). Na Figura 8 temos o caso ideal, onde subdividimos a linha em *buckets* de tamanho s, onde pontos próximos ficam no mesmo *bucket* e pontos distantes ficam em *buckets* diferentes.

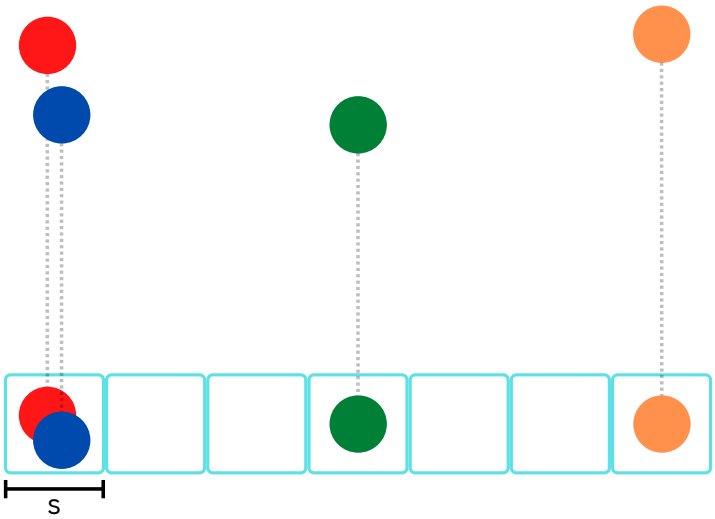


Figura 8: Pontos num espaço bidimensional que tiveram uma boa projeção em uma tabela *hash* ($t = 1$).

Já na Figura 9 podemos observar dois casos não ideais: em (a) temos uma quantização de *buckets* ruim, com a divisão entre *buckets* separando pontos próximos; uma solução seria adicionar mais *buckets*, porém isso faria o cálculo mais computacionalmente exigente. Por outro lado, em (b) temos uma projeção ruim, fazendo com que pontos que no espaço bidimensional são distantes estejam muito próximos quando são projetados na tabela *hash*. Isso demonstra a necessidade de mais de uma linha e, portanto, mais de uma função *hash*.

A família de funções *hash* H que é utilizada para fazer realizar as projeções de distâncias Euclidianas é definida como (Li et al., 2014):

$$h_i(\vec{x}) = \left\lfloor \frac{\vec{x} \cdot \vec{v_i}}{s} \right\rfloor, \quad i = 1, 2, ..., t \tag{5}$$

Aqui, $\vec{x}$ é um ponto no espaço métrico $(\mathcal{D}, d)$, $\vec{v_i}$ é um vetor *hash* gerado no espaço e s é o tamanho do *bucket*.

Assim, para cada ponto, geramos uma projeção para cada um dos t vetores, *hashando* os pontos em um *bucket* para cada vetor. Após fazermos isso com todos os nossos pontos, podemos comparar os itens que estejam dentro do mesmo *bucket* para fazer a busca de similaridade. Voltando ao exemplo bidimensional, o espaço métrico com mais vetores pode ser visualizado na Figura 10.

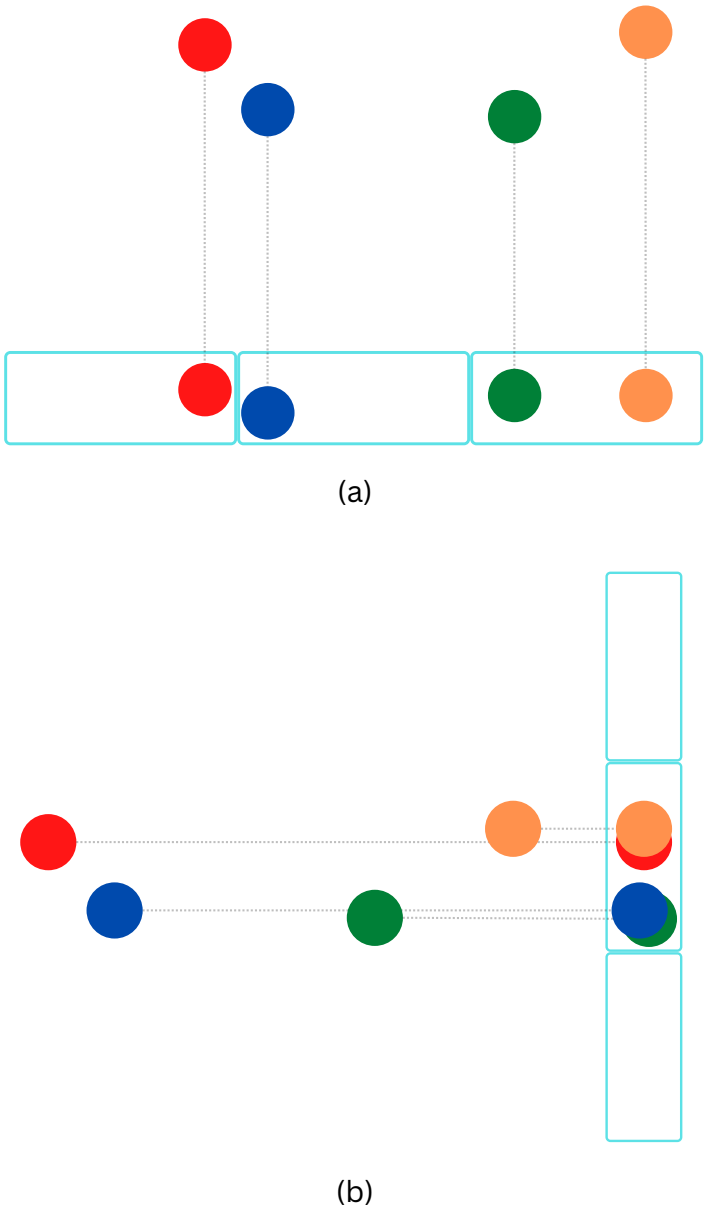


Figura 9: Pontos num espaço bidimensional que tiveram uma projeção ruim. (a) apresenta uma quantização ruim dos *buckets* e (b) apresenta uma projeção para uma tabela *hash* mal orientada com relação aos dados.

Com os pontos *hashados* em t tabelas *hash*, podemos realizar a busca de vizinhos próximos. A partir da chave q, o algoritmo itera sobre as t tabelas *hash*. Para cada tabela, ele calcula a distância dos pontos *hashados* no mesmo *bucket* de q até o ponto q. Após realizar esse

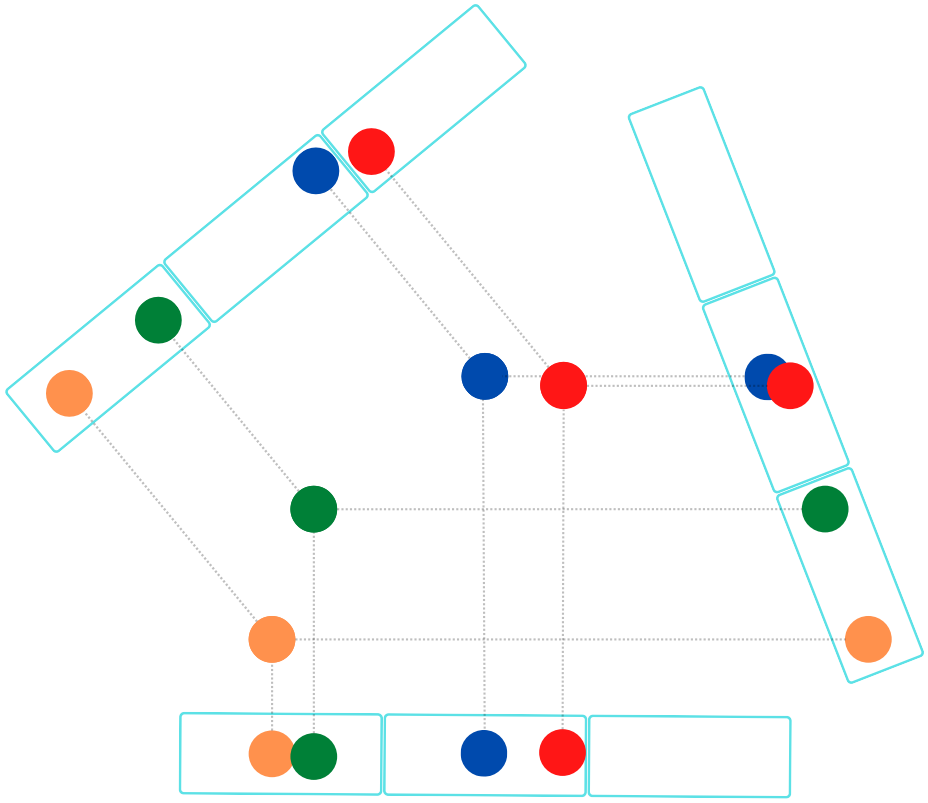


Figura 10: Pontos num espaço bidimensional projetados em várias tabelas *hash* ($t = 3$). Pode-se perceber que projetando em mais tabelas resolvemos os problemas demonstrados na Figura 9.

cálculo, o algoritmo retorna os k pontos que tiveram a menor distância de q, realizando a busca aproximada de vizinhos próximos de uma maneira computacionalmente eficiente, enquanto sendo capaz de capturar os pontos mais próximos de q.

3 Aplicação e Resultados

Neste capítulo apresentaremos os dados que utilizamos para realizar a busca de similaridade de LSBGs com nosso modelo. Além disso, descrevemos como processamos e preparemos esses dados para poder utilizá-los com o LSH, bem como quais catálogos consideramos para realizar testes e analisar a performance do modelo e qualidade de nossos resultados. Por fim, apresentamos as configurações utilizadas para o treinamento do algoritmo e os resultados obtidos.

3.1 Dados

Para nosso trabalho utilizamos dados do *Dark Energy Survey* (DES)[4], uma sondagem fotométrica no óptico que cobre $\sim 5000\text{deg}^2$ do céu usando a *Dark Energy Camera* (DECam) (Flaugher et al., 2015) no Telescópio Blanco, localizado no Observatório Inter-Americano Cerro Tololo.

Usamos os dados coletados nos primeiros três anos de observação do DES (DES Y3) (Abbott et al., 2022). Esses dados apresentam as mesmas técnicas de processamento de imagens e detecção de objetos utilizada no primeiro lançamento de dados do DES (Abbott et al., 2018). Nota-se que a detecção de objetos foi realizada em imagens $r + i + z$ com a ferramenta `SourceExtractor` (Bertin, 2006). O processo de detecção do DES foi otimizado para a detecção e medida de galáxias a distâncias cosmológicas, que geralmente apresentam brilho superficial fraco e um tamanho relativamente pequeno.

Nos baseamos no trabalho de Tanoglidis et al. (2021a) para selecionar uma amostra de objetos astronômicos que apresentam propriedades similares a LSBGs a partir das propriedades físicas deles. Esses dados mais especializados são os que usamos para a nossa busca de LSBGs por similaridade. Assim, partimos do catálogo *DES Y3 Gold coadd (v2.2)*, obtido de detecções do `SourceExtractor` (Sevilla-Noarbe et al., 2021). Realizamos então vários cortes nos parâmetros do catálogo para remover objetos que claramente não são LSBGs, sem correções sendo aplicadas nessa fase.

Seguindo os cortes de Tanoglidis et al. (2021a) e baseados nas recomendações de uso do catálogo Greco et al. (2018) *DES Y3 Gold coadd (v2.2)*, selecionamos objetos nos baseando no tamanho angular e brilho superficial dos objetos. Começamos com a remoção de objetos pontuais. Também, consideramos somente fontes que têm raio efetivo na banda g no intervalo $2.5'' < r_{1/2}(g) < 20''$ e brilho superficial médio dentro do raio efetivo no intervalo $24.2 < \bar{\mu}_{\text{eff}}(g) < 28.8$ mag arcsec^{-2}. Realizamos cortes baseados na cor, guiados pela análise de Greco et al. (2018) e realizamos os seguintes cortes: $-0.1 < g-i < 1.4$, $(g-r) > 0.7\cdot(g-i)-0.4$ e $(g-r) < 0.7 \cdot (g-i) + 0.4$. Finalmente, também forçamos que os objetos na nossa amostra apresentem elipticidade menor que 0.7 a fim de eliminar artefatos de alta elipticidade.

Para remover objetos pontuais, consideramos a banda i do parâmetro `SPREAD_MODEL`,

[4] `https://www.darkenergysurvey.org/`

que é um classificador morfológico onde valores próximos a 0 equivalem a estrelas e valores maiores correspondem a galáxias. Também utilizamos o parâmetro `EXTENDED_CLASS_COADD`, que é um classificador entre estrelas de alta confiança (0), estrelas candidatas (1), galáxias candidatas (2) e galáxias de alta confiança (3). Para os cortes de raio utilizamos o parâmetro `FLUX_RADIUS_G`, que é o raio efetivo do objeto na banda g, e para os cortes de brilho superficial utilizamos `MU_MEAN_MODEL_G`, que é o brilho superficial médio na banda g do objeto.

Para os cortes de cor, utilizamos os parâmetros `MAG_AUTO_{G,I,R}`, que são as medidas de magnitude nas bandas g, i e r, respectivamente.

Para os cortes de elipticidade, usamos o parâmetro `A_IMAGE` (semieixo maior) e `B_IMAGE` (semieixo menor). Além disso, removemos objetos que apresentam valores inexistentes em certos parâmetros. A chamada em formato `SQL` realizada no portal do DES[5] com todos os critérios de seleção utilizados é apresentada a seguir:

```
SELECT
    *
FROM
    Y3_GOLD
WHERE
    -- remover fontes pontuais
    EXTENDED_CLASS_COADD != 0 and
    SPREAD_MODEL_I + 5/3*SPREADERR_MODEL_I > 0.007 and
    -- cortes de raio e brilho superficial
    (FLUX_RADIUS_G between 2.5 and 20) and
    (MU_MEAN_MODEL_G between 24.2 and 28.8) and
    -- corte de elipticidade
    ((1 - B_IMAGE/A_IMAGE) < 0.7) and
    -- cortes de cor
    (MAG_AUTO_G-MAG_AUTO_I between -0.1 and 1.4) and
    (MAG_AUTO_G - MAG_AUTO_R) > 0.7*(MAG_AUTO_G - MAG_AUTO_I) - 0.4 and
    (MAG_AUTO_G - MAG_AUTO_R) < 0.7*(MAG_AUTO_G - MAG_AUTO_I) + 0.4 and
    -- remover itens sem valor
    MOF_FLAGS is not NULL and
    SOF_FLAGS is not NULL and
    EXTENDED_CLASS_COADD != -9 and
    EXTENDED_CLASS_MASH_SOF != -9 and
    EXTENDED_CLASS_MASH_MOF != -9 and
    EXTENDED_CLASS_MOF != -9 and
    EXTENDED_CLASS_SOF != -9 and
    EXTENDED_CLASS_WAVG != -9
```

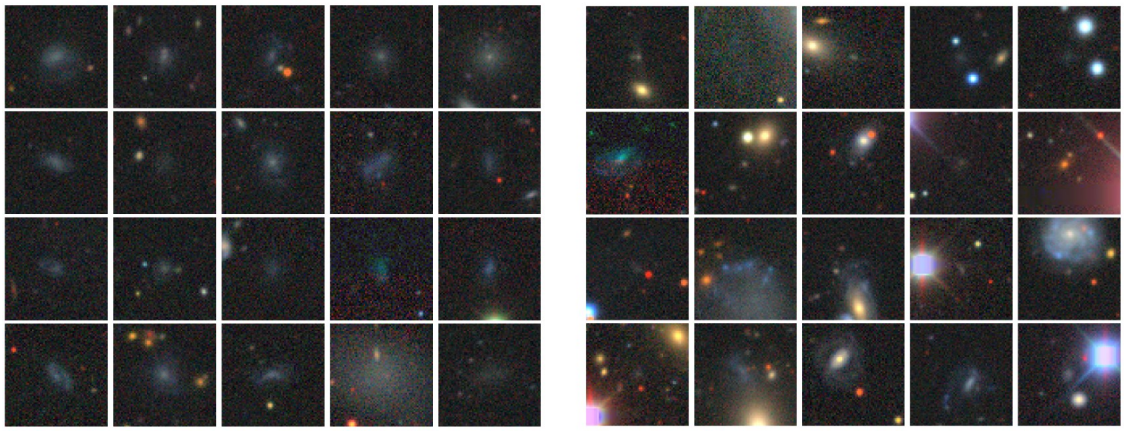
Após a realização desses cortes, removemos todas as colunas relacionadas com coordenadas,

[5] `https://www.darkenergysurvey.org/`

exceto pela ascensão reta e declinação. Isso é feito porque a localização do objeto não traz informações sobre suas propriedades e pode influenciar nosso modelo negativamente. Com isso, nossa amostra final apresenta 11670190 objetos e 370 colunas, de uma amostra inicial do catálogo inteiro que continha $\sim$ 400 milhões de objetos.

Para testar o LSH utilizamos a amostra de 23790 LSBGs obtida por Tanoglidis et al. (2021a). Após realizar os mesmos cortes descritos anteriormente de propriedades físicas e remoção de itens sem valor, essa amostra cai para 18685 LSBGs, que são os objetos que utilizaremos como chaves para realizar nossas buscas e testar a performance das mesmas. A diferença no número de objetos dessas amostra se deve ao fato de termos aplicado critérios de seleção de objetos no catálogo do DES mais restritivos em relação à qualidade dos dados do que os feitos por Tanoglidis et al. (2021a).

Também utilizamos duas amostras de artefatos obtidas por Tanoglidis et al. (2021b) a fim de testar uma possível aplicação da busca de vizinhos próximos para identificar itens em catálogos que não são objetos celestes reais, mas sim artefatos observacionais. A amostra de artefatos é composta de objetos que foram marcados por métodos paramétricos e de aprendizado de máquina como sendo LSBGs, mas que após inspeção visual, foram considerados como não fazendo parte dessa população. Artefatos do tipo 1 são dominados pela presença de fortes rastros de difração e são mais facilmente detectáveis, enquanto que artefatos do tipo 2 são mais sutis, havendo sido marcados pelo método de aprendizado de máquina em dados tabulares como LSBGs incorretamente. Uma comparação das LSBGs e dos artefatos é feita na Figura 11.



(a) (b)

Figura 11: Objetos classificados pelo modelo de Tanoglidis et al. (2021a) como LSBGs. (a) objetos classificados como LSBGs pela análise manual; (b) objetos classificados como artefatos pela análise manual.

Fonte: Tanoglidis et al. (2021a)

3.2 Processamento dos dados

Tendo os dados, aplicamos técnicas padrões de pré-processamento de aprendizado de máquina para prepará-los para a busca de vizinhos próximos. Para processamento de dados e busca de vizinhos próximos utilizamos a biblioteca *PySpark*[6], uma interface de *Apache Spark*[7] para a linguagem de programação *Python*. A ferramenta *Apache Spark* é uma ferramenta para processamento de dados em grande de escala, que permite o paralelismo implícito com *clusters* computacionais.

Inicialmente, removemos todos os parâmetros marcados como `FLAGS`. Esses parâmetros associados à qualidade dos dados apresentam códigos que identificam possíveis objetos problemáticos ou anomalias. Como essas colunas não apresentam propriedades físicas, foram removidas. Após esses cortes ficamos com 354 colunas. Uma delas é o `ID` dos objetos, que, por ser utilizado somente para a identificação de objetos, não é utilizada como parâmetro para nosso modelo. As outras duas colunas representam a acensão reta e a declinação que, por serem coordenadas, são somente modos de localizar objetos, novamente não apresentando alguma propriedade física inata. Assim, após remover essas três colunas, obtemos que o número de parâmetros que será utilizado para a busca de vizinhos próximos é 351.

Podemos dividir os parâmetros entre reais e categóricos. Valores reais são números reais, sendo associados com medidas físicas ou o erro dessas medidas. Já os parâmetros categóricos apresentam um conjunto discreto de valores possíveis usados para representar diferentes categorias do parâmetro. Um exemplo de parâmetro dessa categoria é o `EXTENDED_CLASS_COADD` (discutido na seção 3.1). Se nesse parâmetro não for realizado nenhum processamento, o modelo de busca de vizinhos próximos pode entender que itens marcados como galáxias de alta confiança (3) são maiores que itens marcados como galáxias candidatas (2) pois $3 > 2$ (Raschka and Mirjalili, 2019).

Assim, utilizamos a técnica *One-Hot Encoding* (OHE) (Raschka and Mirjalili, 2019), que consiste em criar parâmetros (colunas) novos binários para os parâmetros categóricos. Se um parâmetro pode obter valores diferentes, como no caso do `EXTENDED_CLASS_COADD`, criamos parâmetros novos em seu lugar. Cada um desses parâmetros pode possuir o valor 0 ou 1 e equivale a um dos valores possíveis. Se um objeto, por exemplo, apresentar `EXTENDED_CLASS_COADD` $= 2$, somente o parâmetro que equivale ao valor 2 fica ativado, com os outros parâmetros sendo iguais a zero. Um exemplo simplificado dessa técnica é realizado na Figura 12. Aplicamos OHE em 6 colunas categóricas do nosso catálogo.

Havendo processado os dados categóricos, realizamos uma normalização de todos os nossos dados (parâmetros numéricos e categóricos). Isso é feito para facilitar as medidas de distância no nosso algoritmo de LSH. Ao normalizar todos os parâmetros, os valores ficam na mesma escala, fazendo com que possamos utilizar todos os parâmetros para obter informações importantes sobre os dados, sem um parâmetro ter uma prioridade maior simplesmente por estar em outra escala (Raschka and Mirjalili, 2019). Realizamos a normalização aplicando a

[6]https://spark.apache.org/docs/latest/api/python/
[7]https://spark.apache.org/

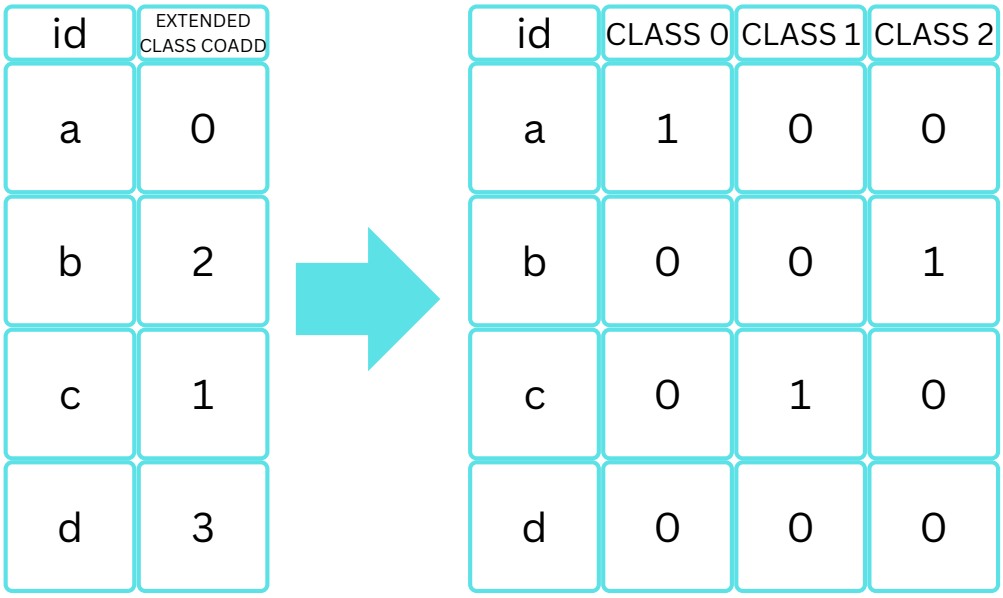


Figura 12: Exemplo de OHE sendo utilizado na coluna `EXTENDED_CLASS_COADD`.

equação:

$$z_i = \frac{x_i - \mu}{\sigma} \,, \tag{6}$$

onde z_i é o valor normalizado do parâmetro x_i para o i-ésimo objeto no nosso conjunto de dados, x_i é o parâmetro x do i-ésimo objeto de nosso conjunto de dados, μ é a média do parâmetro x de todos os objetos do nosso conjunto de dados e σ é o desvio padrão do parâmetro x de todos os objetos do nosso conjunto de dados.

Com isso nossos 11670190 objetos estão processados e preparados para a busca de vizinhos próximos aproximada utilizando o LSH.

3.3 Treinamento do algoritmo

Para realizar a busca, utilizamos a implementação do algoritmo LSH disponível na biblioteca *PySpark*. Mais especificamente, utilizamos a classe `BucketedRandomProjectionLSH`, uma família LSH para a distância Euclidiana (vide Equação 2).

Nessa implementação, o modelo de vizinhos próximos apresenta dois hiperparâmetros (parâmetros do modelo) que devem ser definidos pelo usuário:

- s (tamanho dos *buckets*): quanto menor, mais espaços temos para a projeção dos pontos. Podemos pensar no caso limite onde s é muito próximo de zero, que geraria *buckets* do tamanho dos pontos, o que gera maior precisão, mas maior exigência computacional.
- t (número de tabelas *hash*): quanto maior, mais projeções teremos, fazendo um sistema

mais robusto e com menores chances de acabar com pontos distantes em *buckets* similares, porém a exigência computacional se torna maior.

Para determinar os valores ideais para esses hiperparâmetros é necessário a realização de testes a fim de encontrar um acordo entre precisão e eficiência computacional. Porém, não foi possível realizar testes sistemáticos para analisar quais os melhores hiperparâmetros. Após testar alguns conjuntos de valores de s e t e realizar uma análise visual dos vizinhos próximos de diferentes chaves, encontramos $s = 2.0$ e $t = 3$ como valores que retornam bons resultados.

Executamos a busca de vizinhos próximos utilizando um cluster *Amazon EMR*[8] composto de 1 nodo *master* `c5.xlarge` e 1 nodo *core* `c5.4xlarge`. Nessa configuração, o LSH leva menos de um segundo para ser treinado e em torno de 6 minutos para retornar os 25 mil vizinhos mais próximos de uma chave.

O código que utilizamos para nosso modelo está disponível em `https://github.com/zysymu/lsh-astro`.

3.4 Resultados

Para testar a performance de nosso modelo, o ideal seria saber os valores reais das distâncias de um objeto com relação aos outros no espaço métrico. Porém, pelo tamanho dos nossos dados, a obtenção de medidas absolutas para essas distâncias é inviável e assim, nossos resultados serão baseados na semelhança visual dos vizinhos com relação à chave. Utilizamos o catálogo de LSBGs de Tanoglidis et al. (2021a) para escolher algumas chaves para teste e checar a quantidade de vizinhos daquelas chaves que já estão catalogados.

Escolhemos aleatoriamente 10 LSBGs desse catálogo para usar como chaves e buscamos os 25000 vizinhos mais próximos delas. Na Figura 13 podemos ver uma dessas chaves (`ID=157441790`) e, na linha do topo, os 5 vizinhos mais próximos da chave, na linha do meio os 5 vizinhos localizados na metade da nossa busca e na linha de baixo, os 5 vizinhos mais distantes da nossa busca. Marcados em vermelho estão os objetos não presentes no catálogo de LSBGs de Tanoglidis et al. (2021a) e marcados em verde, temos os objetos presentes nesse mesmo catálogo. No Apêndice temos figuras desse mesmo formato com as outras 9 chaves escolhidas aleatoriamente.

Nota-se que os objetos mais similares apresentam morfologia e cor similar à chave. A busca de similaridade pode beneficiar a busca de objetos astronômicos de diversos modos, tendo em vista que somente é necessário um objeto de determinada classe para encontrar k objetos similares.

Na Figura 14 porém, vemos um histograma do número de objetos retornados na busca por similaridade com relação às suas distâncias. Nela também consideramos os objetos retornados que estão presentes no catálogo de LSBGs de Tanoglidis et al. (2021a) e apresentamos o histograma destes. Podemos notar que nesse caso por exemplo, com somente uma galáxia de baixo brilho superficial (nossa chave), conseguimos encontrar 12260 galáxias da amostra

[8] https://aws.amazon.com/emr/

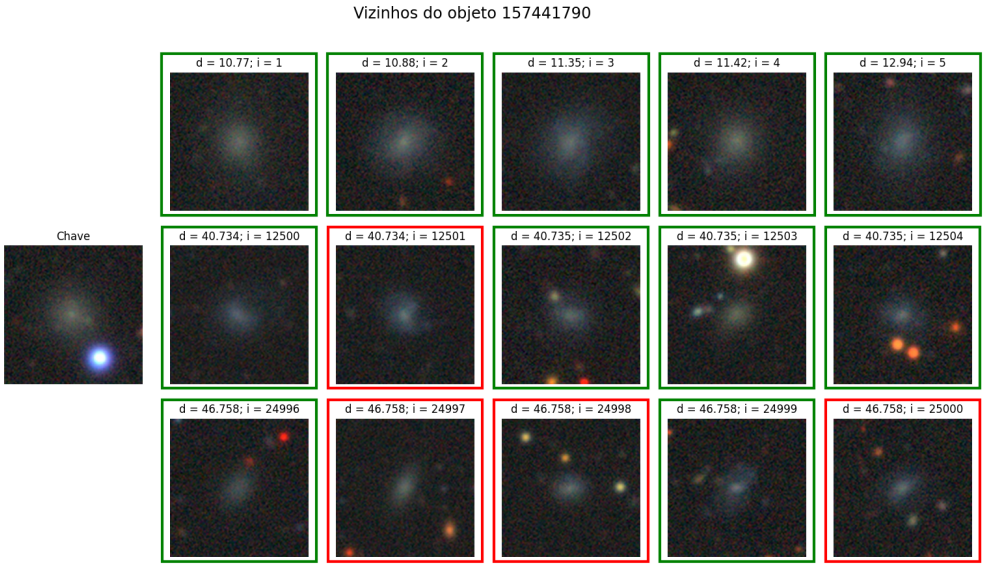


Figura 13: Vizinhos mais próximos da LSBG de `ID=157441790`, onde `d` é a distância à chave e `i` é posição. Na linha do topo, os 5 vizinhos mais próximos da chave, na linha do meio os 5 vizinhos localizados na metade da nossa busca e na linha de baixo, os 5 vizinhos mais distantes na nossa busca. Marcados em vermelho estão os objetos não presentes no catálogo de LSBGs de Tanoglidis et al. (2021a) e marcados em verde, temos os objetos presentes nesse mesmo catálogo.

de Tanoglidis et al. (2021a) entre os 25000 vizinhos mais próximos da nossa chave. Isso corresponde a 49.04% dos vizinhos retornados sendo LSBGs já catalogadas. Outros objetos, principalmente os próximos à chave são visualmente muito similares a LSBGs, sendo possíveis objetos dessa classe que não estão presentes no catálogo. Encontramos objetos do catálogo de Tanoglidis et al. (2021a) mesmo entre os últimos dos 25000 vizinhos mais próximos.

Notamos assim uma das possíveis aplicações desse método: o seu uso como um refinador. Ao realizar uma busca para um valor grande de k, é possível obter vários objetos similares à chave. Isso permite a diminuição do número de dados que precisam ser analisados por técnicas paramétricas computacionalmente custosas ou classificados por redes neurais convolucionais, dado que já se é possível obter uma grande amostra de dados com as mesmas propriedades automaticamente havendo somente uma chave. Nota-se que esses novos objetos são visualmente muito similares às LSBGs de Tanoglidis et al. (2021a), entretanto, para confirmar se são de fato LSBGs seria necessário um estudo mais detalhado das suas propriedades físicas através de métodos paramétricos.

Para demonstrar a generalidade de nosso modelo, realizamos também buscas dos 25000 vizinhos mais próximos para artefatos. Escolhemos aleatoriamente 5 artefatos do tipo 1 e 5 artefatos do tipo 2 como chaves. Na Figura 15 podemos ver os objetos retornados para nossa

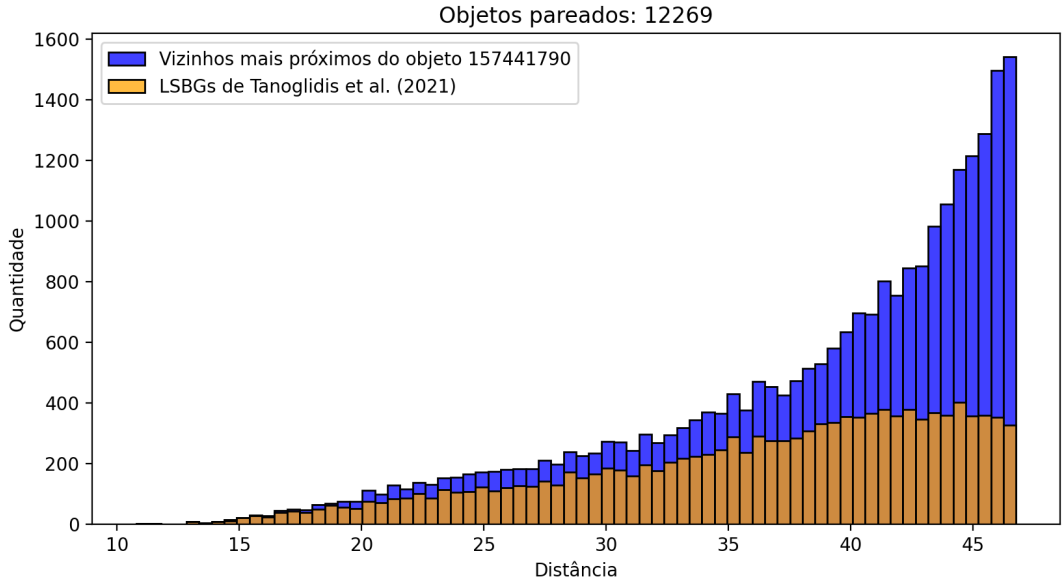


Figura 14: Em azul, histograma da distância dos 25000 vizinhos mais próximos do objeto de `ID=157441790`. Em laranja temos o histograma da distância dos vizinhos mais próximos da mesma chave que estão presentes no catálogo de Tanoglidis et al. (2021a).

busca de similaridade para uma chave de artefato de tipo 1. Já na Figura 16 temos os objetos retornados para uma chave de artefato de tipo 2. Nota-se que em ambos os casos nosso modelo, que é baseado unicamente na propriedade física dos objetos, é capaz de encontrar outras casos com propriedades visuais muito similares. Nota-se também que muitos dos objetos retornados não estão presentes no catálogo de Tanoglidis et al. (2021b). Em parte isso se deve ao fato que muitos cortes que realizamos para obter o catálogo final de objetos a ser analisado podem ter removido alguns artefatos da amostra de Tanoglidis et al. (2021b).

Além disso, reiteramos que o modelo utilizado para a busca de artefatos é o mesmo da busca de LSBGs. Para encontrar objetos similares de qualquer tipo em algum catálogo, após o pré-processamento e treinamento do modelo, somente é necessário uma chave.

Nota-se que esse método não pretende criar amostrar completas e puras. Como é observado na Figura 14, enquanto retornamos vários objetos que são LSBGs, a quantidade de objetos retornada cresce rapidamente com a distância. Além disso, objetos próximos podem ser visualmente similares a LSBGs, mesmo sem ser de fato galáxias desse tipo. Somente através de análise detalhada com métodos paramétricos poderemos verificar se nosso método encontra LSBGs novas que foram perdidas pelo método de Tanoglidis et al. (2021a).

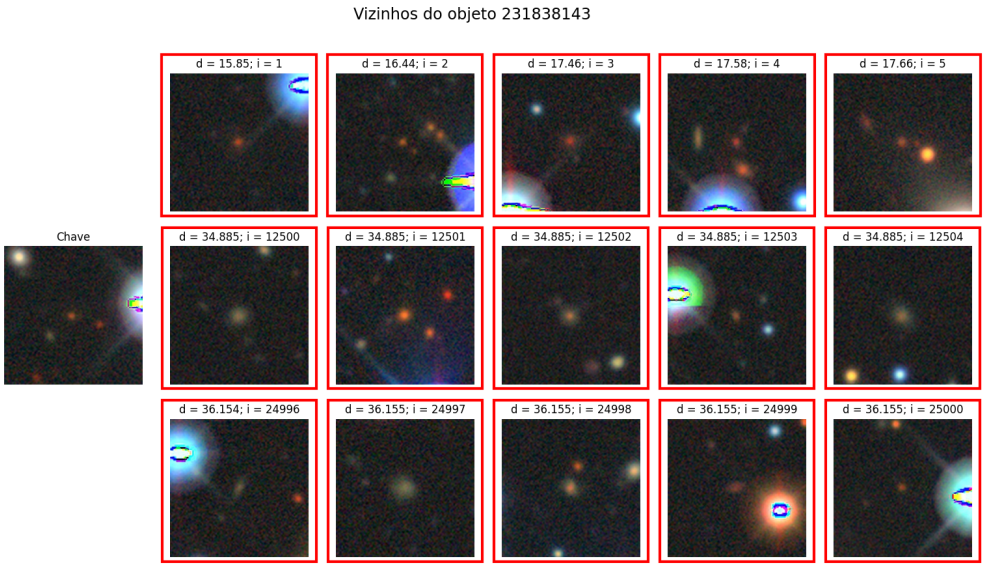


Figura 15: Vizinhos mais próximos do artefato tipo 1 de `ID=231838143`. A estrutura da figura é igual à Figura 13.

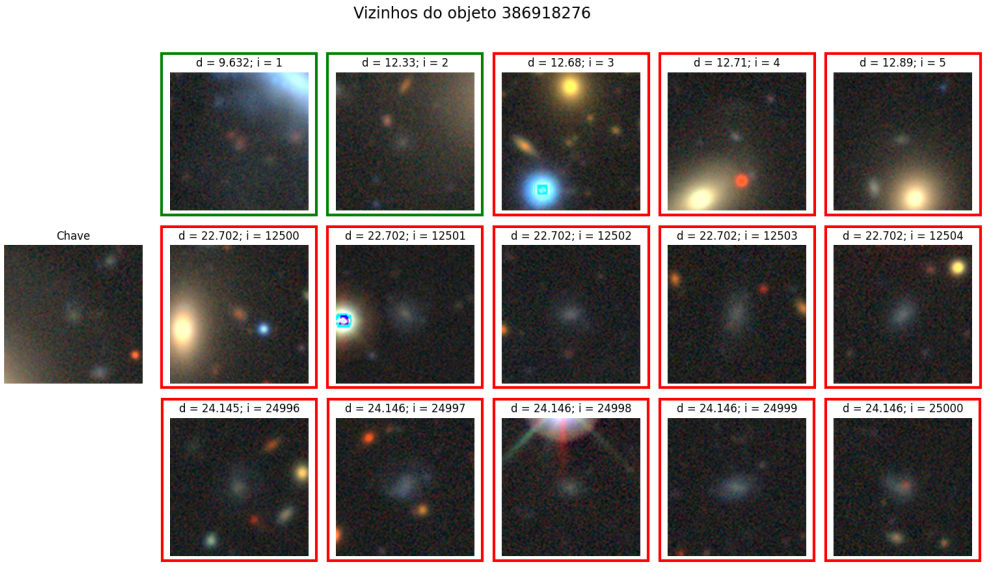


Figura 16: Vizinhos mais próximos do artefato tipo 2 de `ID=386918276`. A estrutura da figura é igual à Figura 13.

4 Conclusão

Com futuros levantamentos fotométricos, como Euclid e LSST, esperados a produzir quantidades enormes de dados, métodos que permitem a descoberta de novos objetos automaticamente são de suma necessidade. Um desses métodos, a busca aproximada de vizinhos próximos permite encontrar objetos similares a uma chave, sendo necessário somente um item de certo tipo para encontrar objetos da mesma classe. Como lidamos com propriedades que apresentam valores reais, utilizamos o método LSH com uma distância Euclidiana.

Além disso, como o LSH permite o uso de dados estruturados de propriedades físicas dos objetos, o que o torna computacionalmente eficiente, sendo possível processar muitos itens em pouco tempo e sem a necessidade de recursos computacionais custosos. Como a busca de objetos similares é feita no espaço métrico, é possível encontrar candidatos espalhados por todo o campo de visão dos levantamentos.

Nesse trabalho abordamos o problema de encontrar LSBGs utilizando a busca de vizinhos aproximada com o LSH. Processamos ~ 400 milhões de dados do catálogo *DES Y3 Gold coadd (v2.2)*, realizando cortes baseados em propriedades físicas para obter uma amostra de 11670190 objetos que condizem com as propriedades gerais de LSBGs.

Utilizando a ferramenta *Apache Spark* para o processamento de grandes quantidades de dados estruturados em paralelo, preparamos os dados, excluindo propriedades consideradas não significativas e transformando tanto propriedades numéricas quanto categóricas para um formato que permite maior eficiência do algoritmo de LSH. Após isso, aplicamos a busca de vizinhos aproximada com LSBGs e artefatos conhecidos para checar a performance do nosso modelo.

Encontramos objetos muito similares às chaves presentes em catálogos e outros objetos visualmente muito similares não presentes. No caso de LSBGs, encontramos diversos objetos não presentes no catálogo de Tanoglidis et al. (2021a), que analisou a mesma amostra observada no nosso trabalho, mas que visualmente são fortes candidatos a galáxias de baixo brilho superficial. Já no caso de artefatos, também podemos encontrar itens que apresentam propriedades similares, demonstrando a generalidade do método e também como o mesmo pode ser utilizado para limpar amostras.

Assim, esse trabalho apresenta uma alternativa para encontrar diferentes tipos de objetos astronômicos de um modo altamente eficiente, tanto de um ponto de vista de tamanho da amostra quanto de processamento e tempo computacionais. Acreditamos que a técnica aqui apresentada pode ser utilizada em levantamentos astronômicos para a descoberta de candidatos para os mais diferentes tipos de objetos.

4.1 Próximos passos

Demonstramos a eficiência e performance do uso de LSH para encontrar objetos astronômicos similares, em especial LSBGs. Para isso utilizamos uma amostra que é somente

uma fração do número de objetos presentes em catálogos astronômicos completos. Assim, seria de interesse em futuros trabalhos analisar o uso de LSH para catálogos inteiros, sem nenhum corte baseado em propriedades físicas. Por extensão, é possível também analisar o quão efetivo é esse tipo de modelo para outros objetos raros.

Uma pesquisa de hiperparâmetros completa e uso de métricas para quantificar como diferentes valores influenciam a performance do modelo também seria de forte interesse. Isso permitiria encontrar o LSH que obtém a melhor performance enquanto é computacionalmente eficiente. A dificuldade desse objetivo é determinar quais são os itens mais próximos dos itens do catálogo. Uma possibilidade é o uso do algoritmo de kNN para encontrar os k vizinhos mais próximos de certos objetos e comparar esses resultados com os vizinhos aproximados do LSH.

Finalmente, com relação ao modelo, é possível testar outras alternativas para busca aproximada de vizinhos próximos. Com diferentes métricas de distância, talvez algo além da distância Euclidiana poderia apresentar uma performance melhor. Além disso métodos recentes que utilizam redes neurais em combinação com a técnica de *hashing* (Moran, 2017) podem ser de interesse para a criação de modelos mais especializados em dados astronômicos.

5 Referências

Abbott, T., Aguena, M., Alarcon, A., Allam, S., Alves, O., Amon, A., Andrade-Oliveira, F., Annis, J., Avila, S., Bacon, D., Baxter, E., Bechtol, K., Becker, M., Bernstein, G., Bhargava, S., Birrer, S., Blazek, J., Brandao-Souza, A., Bridle, S., Brooks, D., Buckley-Geer, E., Burke, D., Camacho, H., Campos, A., Rosell, A.C., Kind, M.C., Carretero, J., Castander, F., Cawthon, R., Chang, C., Chen, A., Chen, R., Choi, A., Conselice, C., Cordero, J., Costanzi, M., Crocce, M., da Costa, L., da Silva Pereira, M., Davis, C., Davis, T., Vicente, J.D., DeRose, J., Desai, S., Valentino, E.D., Diehl, H., Dietrich, J., Dodelson, S., Doel, P., Doux, C., Drlica-Wagner, A., Eckert, K., Eifler, T., Elsner, F., Elvin-Poole, J., Everett, S., Evrard, A., Fang, X., Farahi, A., Fernandez, E., Ferrero, I., Ferté, A., Fosalba, P., Friedrich, O., Frieman, J., García-Bellido, J., Gatti, M., Gaztanaga, E., Gerdes, D., Giannantonio, T., Giannini, G., Gruen, D., Gruendl, R., Gschwend, J., Gutierrez, G., Harrison, I., Hartley, W., Herner, K., Hinton, S., Hollowood, D., Honscheid, K., Hoyle, B., Huff, E., Huterer, D., Jain, B., James, D., Jarvis, M., Jeffrey, N., Jeltema, T., Kovacs, A., Krause, E., Kron, R., Kuehn, K., Kuropatkin, N., Lahav, O., Leget, P.F., Lemos, P., Liddle, A., Lidman, C., Lima, M., Lin, H., MacCrann, N., Maia, M., Marshall, J., Martini, P., McCullough, J., Melchior, P., Mena-Fernández, J., Menanteau, F., Miquel, R., Mohr, J., Morgan, R., Muir, J., Myles, J., Nadathur, S., Navarro-Alsina, A., Nichol, R., Ogando, R., Omori, Y., Palmese, A., Pandey, S., Park, Y., Paz-Chinchón, F., Petravick, D., Pieres, A., Malagón, A.P., Porredon, A., Prat, J., Raveri, M., Rodriguez-Monroy, M., Rollins, R., Romer, A., Roodman, A., Rosenfeld, R., Ross, A., Rykoff, E., Samuroff, S., Sánchez, C., Sanchez, E., Sanchez, J., Cid, D.S., Scarpine, V., Schubnell, M., Scolnic, D., Secco, L., Serrano, S., Sevilla-Noarbe, I., Sheldon, E., Shin, T., Smith, M., Soares-Santos, M., Suchyta, E., Swanson, M., Tabbutt, M., Tarle, G., Thomas, D., To, C., Troja, A., Troxel, M., Tucker, D., Tutusaus, I., Varga, T., Walker, A., Weaverdyck, N., Wechsler, R., Weller, J., Yanny, B., Yin, B., Zhang, Y., and, J.Z., 2022. Dark energy survey year 3 results: Cosmological constraints from galaxy clustering and weak lensing. Physical Review D 105. URL: `https://doi.org/10.1103%2Fphysrevd.105.023520`, doi:`10.1103/physrevd.105.023520`.

Abbott, T.M.C., Abdalla, F.B., Allam, S., Amara, A., Annis, J., Asorey, J., Avila, S., Ballester, O., Banerji, M., Barkhouse, W., Baruah, L., Baumer, M., Bechtol, K., Becker, M.R., Benoit-Lévy, A., Bernstein, G.M., Bertin, E., Blazek, J., Bocquet, S., Brooks, D., Brout, D., Buckley-Geer, E., Burke, D.L., Busti, V., Campisano, R., Cardiel-Sas, L., Rosell, A.C., Kind, M.C., Carretero, J., Castander, F.J., Cawthon, R., Chang, C., Chen, X., Conselice, C., Costa, G., Crocce, M., Cunha, C.E., D'Andrea, C.B., Costa, L.N.d., Das, R., Daues, G., Davis, T.M., Davis, C., Vicente, J.D., DePoy, D.L., DeRose, J., Desai, S., Diehl, H.T., Dietrich, J.P., Dodelson, S., Doel, P., Drlica-Wagner, A., Eifler, T.F., Elliott, A.E., Evrard, A.E., Farahi, A., Neto, A.F., Fernandez, E., Finley, D.A., Flaugher, B., Foley, R.J., Fosalba, P., Friedel, D.N., Frieman, J., García-Bellido, J., Gaztanaga, E., Gerdes, D.W., Giannantonio, T., Gill, M.S.S., Glazebrook, K., Goldstein, D.A., Gower, M., Gruen, D., Gruendl, R.A., Gschwend, J., Gupta, R.R., Gutierrez, G., Hamilton, S., Hartley,

W.G., Hinton, S.R., Hislop, J.M., Hollowood, D., Honscheid, K., Hoyle, B., Huterer, D., Jain, B., James, D.J., Jeltema, T., Johnson, M.W.G., Johnson, M.D., Kacprzak, T., Kent, S., Khullar, G., Klein, M., Kovacs, A., Koziol, A.M.G., Krause, E., Kremin, A., Kron, R., Kuehn, K., Kuhlmann, S., Kuropatkin, N., Lahav, O., Lasker, J., Li, T.S., Li, R.T., Liddle, A.R., Lima, M., Lin, H., López-Reyes, P., MacCrann, N., Maia, M.A.G., Maloney, J.D., Manera, M., March, M., Marriner, J., Marshall, J.L., Martini, P., McClintock, T., McKay, T., McMahon, R.G., Melchior, P., Menanteau, F., Miller, C.J., Miquel, R., Mohr, J.J., Morganson, E., Mould, J., Neilsen, E., Nichol, R.C., Nogueira, F., Nord, B., Nugent, P., Nunes, L., Ogando, R.L.C., Old, L., Pace, A.B., Palmese, A., Paz-Chinchón, F., Peiris, H.V., Percival, W.J., Petravick, D., Plazas, A.A., Poh, J., Pond, C., Porredon, A., Pujol, A., Refregier, A., Reil, K., Ricker, P.M., Rollins, R.P., Romer, A.K., Roodman, A., Rooney, P., Ross, A.J., Rykoff, E.S., Sako, M., Sanchez, M.L., Sanchez, E., Santiago, B., Saro, A., Scarpine, V., Scolnic, D., Serrano, S., Sevilla-Noarbe, I., Sheldon, E., Shipp, N., Silveira, M.L., Smith, M., Smith, R.C., Smith, J.A., Soares-Santos, M., Sobreira, F., Song, J., Stebbins, A., Suchyta, E., Sullivan, M., Swanson, M.E.C., Tarle, G., Thaler, J., Thomas, D., Thomas, R.C., Troxel, M.A., Tucker, D.L., Vikram, V., Vivas, A.K., Walker, A.R., Wechsler, R.H., Weller, J., Wester, W., Wolf, R.C., Wu, H., Yanny, B., Zenteno, A., Zhang, Y., Zuntz, J., Juneau, S., Fitzpatrick, M., Nikutta, R., Nidever, D., Olsen, K., Scott, A., and, 2018. The Dark Energy Survey: Data Release 1. ApJS 239, 18. URL: https://doi.org/10.3847/1538-4365/aae9f0, doi:10.3847/1538-4365/aae9f0. publisher: American Astronomical Society.

Alexander, S., Gleyzer, S., Reddy, P., Tidball, M., Toomey, M.W., 2021. Domain Adaptation for Simulation-Based Dark Matter Searches Using Strong Gravitational Lensing. URL: http://arxiv.org/abs/2112.12121, doi:10.48550/arXiv.2112.12121. arXiv:2112.12121 [astro-ph].

Amorisco, N.C., Monachesi, A., Agnello, A., White, S.D.M., 2018. The globular cluster systems of 54 Coma ultra-diffuse galaxies: statistical constraints from HST data. Monthly Notices of the Royal Astronomical Society 475, 4235–4251. URL: https://doi.org/10.1093/mnras/sty116, doi:10.1093/mnras/sty116.

Beasley, M.A., Romanowsky, A.J., Pota, V., Navarro, I.M., Martinez Delgado, D., Neyer, F., Deich, A.L., 2016. An Overmassive Dark Halo around an Ultra-diffuse Galaxy in the Virgo Cluster. 819, L20. doi:10.3847/2041-8205/819/2/L20, arXiv:1602.04002.

Bertin, E., 2006. Automatic Astrometric and Photometric Calibration with SCAMP 351, 112. URL: https://ui.adsabs.harvard.edu/abs/2006ASPC..351..112B. conference Name: Astronomical Data Analysis Software and Systems XV ADS Bibcode: 2006ASPC..351..112B.

Bhatia, N., Author, C., 2010. Survey of Nearest Neighbor Techniques 8, 4.

Bom, C.R., Fraga, B.M.O., Dias, L.O., Schubert, P., Valentin, M.B., Furlanetto, C., Makler, M., Teles, K., de Albuquerque, M.P., Metcalf, R.B., 2022. Developing a Victorious Strategy

to the Second Strong Gravitational Lensing Data Challenge. Monthly Notices of the Royal Astronomical Society 515, 5121–5134. URL: http://arxiv.org/abs/2203.09536, doi:10.1093/mnras/stac2047. arXiv:2203.09536 [astro-ph].

Bothun, G., Impey, C., McGaugh, S., 1997. Low-Surface-Brightness Galaxies: Hidden Galaxies Revealed. Publications of the Astronomical Society of the Pacific 109, 745–758. URL: https://ui.adsabs.harvard.edu/abs/1997PASP..109..745B, doi:10.1086/133941. aDS Bibcode: 1997PASP..109..745B.

Bothun, G.D., Impey, C.D., Malin, D.F., Mould, J.R., 1987. Discovery of a Huge Low-Surface-Brightness Galaxy: A Proto-Disk Galaxy at Low Redshift? The Astronomical Journal 94, 23. URL: https://ui.adsabs.harvard.edu/abs/1987AJ.....94...23B, doi:10.1086/114443. aDS Bibcode: 1987AJ.....94...23B.

Carlsten, S.G., Greene, J.E., Greco, J.P., Beaton, R.L., Kado-Fong, E., 2021. Structures of dwarf satellites of milky way-like galaxies: Morphology, scaling relations, and intrinsic shapes. The Astrophysical Journal 922, 267. URL: https://doi.org/10.3847%2F1538-4357%2Fac2581, doi:10.3847/1538-4357/ac2581.

Conselice, C.J., 2018. Ultra-diffuse Galaxies Are a Subset of Cluster Dwarf Elliptical/Spheroidal Galaxies. Res. Notes AAS 2, 43. URL: https://doi.org/10.3847/2515-5172/aab7f6, doi:10.3847/2515-5172/aab7f6. publisher: American Astronomical Society.

Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C., 2009. Introduction to Algorithms, Third Edition. 3rd ed., The MIT Press.

Cunningham, P., Delany, S.J., 2020. k-Nearest Neighbour Classifiers: 2nd Edition (with Python examples) URL: https://arxiv.org/abs/2004.04523v2, doi:10.1145/3459665.

Datar, M., Immorlica, N., Indyk, P., Mirrokni, V.S., 2004. Locality-sensitive hashing scheme based on p-stable distributions, in: Proceedings of the Twentieth Annual Symposium on Computational Geometry, Association for Computing Machinery, New York, NY, USA. p. 253–262. URL: https://doi.org/10.1145/997817.997857, doi:10.1145/997817.997857.

Di Cintio, A., Brook, C.B., Dutton, A.A., Macciò, A.V., Obreja, A., Dekel, A., 2017. NIHAO - XI. Formation of ultra-diffuse galaxies by outflows. Monthly Notices of the Royal Astronomical Society 466, L1–L6. URL: https://ui.adsabs.harvard.edu/abs/2017MNRAS.466L...1D, doi:10.1093/mnrasl/slw210. aDS Bibcode: 2017MNRAS.466L...1D.

Disney, M.J., 1976. Visibility of galaxies. Nature 263, 573–575. URL: https://ui.adsabs.harvard.edu/abs/1976Natur.263..573D, doi:10.1038/263573a0. aDS Bibcode: 1976Natur.263..573D.

van Dokkum, P., Danieli, S., Cohen, Y., Merritt, A., Romanowsky, A.J., Abraham, R., Brodie, J., Conroy, C., Lokhorst, D., Mowla, L., O'Sullivan, E., Zhang, J., 2018. A galaxy lacking dark matter. Nature 555, 629–632. URL: https://www.nature.com/articles/

nature25767, doi:10.1038/nature25767. number: 7698 Publisher: Nature Publishing Group.

van Dokkum, P.G., Romanowsky, A.J., Abraham, R., Brodie, J.P., Conroy, C., Geha, M., Merritt, A., Villaume, A., Zhang, J., 2015. SPECTROSCOPIC CONFIRMATION OF THE EXISTENCE OF LARGE, DIFFUSE GALAXIES IN THE COMA CLUSTER. The Astrophysical Journal 804, L26. URL: https://doi.org/10.1088/2041-8205/804/1/l26, doi:10.1088/2041-8205/804/1/l26.

Dokkum, P.G.v., Abraham, R., Merritt, A., Zhang, J., Geha, M., Conroy, C., 2015. FORTY-SEVEN MILKY WAY-SIZED, EXTREMELY DIFFUSE GALAXIES IN THE COMA CLUSTER. ApJL 798, L45. URL: https://doi.org/10.1088/2041-8205/798/2/l45, doi:10.1088/2041-8205/798/2/L45. publisher: American Astronomical Society.

Driver, S.P., 1999. The Contribution of Normal, Dim, and Dwarf Galaxies to the Local Luminosity Density. The Astrophysical Journal 526, L69–L72. URL: https://ui.adsabs.harvard.edu/abs/1999ApJ...526L..69D, doi:10.1086/312379. aDS Bibcode: 1999ApJ...526L..69D.

Flaugher, B., Diehl, H.T., Honscheid, K., Abbott, T.M.C., Alvarez, O., Angstadt, R., Annis, J.T., Antonik, M., Ballester, O., Beaufore, L., Bernstein, G.M., Bernstein, R.A., Bigelow, B., Bonati, M., Boprie, D., Brooks, D., Buckley-Geer, E.J., Campa, J., Cardiel-Sas, L., Castander, F.J., Castilla, J., Cease, H., Cela-Ruiz, J.M., Chappa, S., Chi, E., Cooper, C., da Costa, L.N., Dede, E., Derylo, G., DePoy, D.L., de Vicente, J., Doel, P., Drlica-Wagner, A., Eiting, J., Elliott, A.E., Emes, J., Estrada, J., Neto, A.F., Finley, D.A., Flores, R., Frieman, J., Gerdes, D., Gladders, M.D., Gregory, B., Gutierrez, G.R., Hao, J., Holland, S.E., Holm, S., Huffman, D., Jackson, C., James, D.J., Jonas, M., Karcher, A., Karliner, I., Kent, S., Kessler, R., Kozlovsky, M., Kron, R.G., Kubik, D., Kuehn, K., Kuhlmann, S., Kuk, K., Lahav, O., Lathrop, A., Lee, J., Levi, M.E., Lewis, P., Li, T.S., Mandrichenko, I., Marshall, J.L., Martinez, G., Merritt, K.W., Miquel, R., Muñoz, F., Neilsen, E.H., Nichol, R.C., Nord, B., Ogando, R., Olsen, J., Palaio, N., Patton, K., Peoples, J., Plazas, A.A., Rauch, J., Reil, K., Rheault, J.P., Roe, N.A., Rogers, H., Roodman, A., Sanchez, E., Scarpine, V., Schindler, R.H., Schmidt, R., Schmitt, R., Schubnell, M., Schultz, K., Schurter, P., Scott, L., Serrano, S., Shaw, T.M., Smith, R.C., Soares-Santos, M., Stefanik, A., Stuermer, W., Suchyta, E., Sypniewski, A., Tarle, G., Thaler, J., Tighe, R., Tran, C., Tucker, D., Walker, A.R., Wang, G., Watson, M., Weaverdyck, C., Wester, W., Woods, R., and, B.Y., 2015. THE DARK ENERGY CAMERA. The Astronomical Journal 150, 150. URL: https://doi.org/10.1088/0004-6256/150/5/150, doi:10.1088/0004-6256/150/5/150.

Greco, J.P., Greene, J.E., Strauss, M.A., Macarthur, L.A., Flowers, X., Goulding, A.D., Huang, S., Kim, J.H., Komiyama, Y., Leauthaud, A., Leisman, L., Lupton, R.H., Sifón, C., Wang, S.Y., 2018. Illuminating Low Surface Brightness Galaxies with the Hyper Suprime-Cam Survey. ApJ 857, 104. URL: https://doi.org/10.3847/1538-4357/aab842, doi:10.3847/1538-4357/aab842. publisher: American Astronomical Society.

Hastie, T., Tibshirani, R., Friedman, J., 2001. The Elements of Statistical Learning. Springer Series in Statistics, Springer New York Inc., New York, NY, USA.

Heinermann, J., Kramer, O., Polsterer, K.L., Gieseke, F., 2013. On GPU-Based Nearest Neighbor Queries for Large-Scale Photometric Catalogs in Astronomy, in: Timm, I.J., Thimm, M. (Eds.), KI 2013: Advances in Artificial Intelligence, Springer, Berlin, Heidelberg. pp. 86–97. doi:10.1007/978-3-642-40942-4_8.

Impey, C., Bothun, G., 1997. Low Surface Brightness Galaxies. Annual Review of Astronomy and Astrophysics 35, 267–307. URL: https://doi.org/10.1146/annurev.astro.35.1.267, doi:10.1146/annurev.astro.35.1.267. _eprint: https://doi.org/10.1146/annurev.astro.35.1.267.

Kelley, J.L., 1955. General Topology. D. Van Nostrand. Google-Books-ID: nTIMzgEACAAJ.

Lee, J.H., Kang, J., Lee, M.G., Jang, I.S., 2020. The Nature of Ultra-diffuse Galaxies in Distant Massive Galaxy Clusters: A370 in the Hubble Frontier Fields. 894, 75. doi:10.3847/1538-4357/ab8632, arXiv:2004.01340.

Leskovec, J., Rajaraman, A., Ullman, J.D., 2020. Mining of Massive Datasets. 3 ed., Cambridge University Press. doi:10.1017/9781108684163.

Li, D., Eadie, G.M., Abraham, R.G., Brown, P.E., Harris, W.E., Janssens, S.R., Romanowsky, A.J., van Dokkum, P., Danieli, S., 2022. Light from the Darkness: Detecting Ultra-Diffuse Galaxies in the Perseus Cluster through Over-densities of Globular Clusters with a Log-Gaussian Cox Process. ApJ 935, 3. URL: http://arxiv.org/abs/2204.05487, doi:10.3847/1538-4357/ac7b22. arXiv:2204.05487 [astro-ph, stat].

Li, L., Zhang, Y., Zhao, Y., 2008. k-Nearest Neighbors for automated classification of celestial objects. Science in China Series G: Physics, Mechanics and Astronomy 51, 916–922. URL: https://doi.org/10.1007/s11433-008-0088-4, doi:10.1007/s11433-008-0088-4.

Li, P., Mitzenmacher, M., Shrivastava, A., 2014. Coding for random projections, in: Xing, E.P., Jebara, T. (Eds.), Proceedings of the 31st International Conference on Machine Learning, PMLR, Bejing, China. pp. 676–684. URL: https://proceedings.mlr.press/v32/lie14.html.

Lim, S., Peng, E.W., Côté, P., Sales, L.V., Brok, M.d., Blakeslee, J.P., Guhathakurta, P., 2018. The Globular Cluster Systems of Ultra-diffuse Galaxies in the Coma Cluster. ApJ 862, 82. URL: https://doi.org/10.3847/1538-4357/aacb81, doi:10.3847/1538-4357/aacb81. publisher: American Astronomical Society.

Metcalf, R.B., Meneghetti, M., Avestruz, C., Bellagamba, F., Bom, C.R., Bertin, E., Cabanac, R., Courbin, F., Davies, A., Decencière, E., Flamary, R., Gavazzi, R., Geiger, M., Hartley, P., Huertas-Company, M., Jackson, N., Jullo, E., Kneib, J.P., Koopmans, L.V.E., Lanusse, F., Li, C.L., Ma, Q., Makler, M., Li, N., Lightman, M., Petrillo, C.E., Serjeant, S., Schäfer, C., Sonnenfeld, A., Tagore, A., Tortora, C., Tuccillo, D., Valentín, M.B., Velasco-Forero,

S., Kleijn, G.A.V., Vernardos, G., 2019. The Strong Gravitational Lens Finding Challenge. A&A 625, A119. URL: http://arxiv.org/abs/1802.03609, doi:10.1051/0004-6361/201832797. arXiv:1802.03609 [astro-ph].

Moran, S., 2017. Awesome papers on learning to hash. https://learning2hash.github.io.

Pearson, J., Maresca, J., Li, N., Dye, S., 2021. Strong lens modelling: comparing and combining Bayesian neural networks and parametric profile fitting. Monthly Notices of the Royal Astronomical Society 505, 4362–4382. URL: http://arxiv.org/abs/2103.03257, doi:10.1093/mnras/stab1547. arXiv:2103.03257 [astro-ph].

Prole, D.J., van der Burg, R.F.J., Hilker, M., Davies, J.I., 2019. Observational properties of ultra-diffuse galaxies in low-density environments: field UDGs are predominantly blue and star forming. Monthly Notices of the Royal Astronomical Society 488, 2143–2157. URL: https://ui.adsabs.harvard.edu/abs/2019MNRAS.488.2143P, doi:10.1093/mnras/stz1843. aDS Bibcode: 2019MNRAS.488.2143P.

Raschka, S., Mirjalili, V., 2019. Python Machine Learning, 3rd Ed. Packt Publishing, Birmingham, UK.

Sales, L.V., Navarro, J.F., Peñ afiel, L., Peng, E.W., Lim, S., Hernquist, L., 2020. The formation of ultradiffuse galaxies in clusters. Monthly Notices of the Royal Astronomical Society 494, 1848–1858. URL: https://doi.org/10.1093%2Fmnras%2Fstaa854, doi:10.1093/mnras/staa854.

Sevilla-Noarbe, I., Bechtol, K., Kind, M.C., Rosell, A.C., Becker, M.R., Drlica-Wagner, A., Gruendl, R.A., Rykoff, E.S., Sheldon, E., Yanny, B., Alarcon, A., Allam, S., Amon, A., Benoit-Lévy, A., Bernstein, G.M., Bertin, E., Burke, D.L., Carretero, J., Choi, A., Diehl, H.T., Everett, S., Flaugher, B., Gaztanaga, E., Gschwend, J., Harrison, I., Hartley, W.G., Hoyle, B., Jarvis, M., Johnson, M.D., Kessler, R., Kron, R., Kuropatkin, N., Leistedt, B., Li, T.S., Menanteau, F., Morganson, E., Ogando, R.L.C., Palmese, A., Paz-Chinchón, F., Pieres, A., Pond, C., Rodriguez-Monroy, M., Smith, J.A., Stringer, K.M., Troxel, M.A., Tucker, D.L., Vicente, J.d., Wester, W., Zhang, Y., Abbott, T.M.C., Aguena, M., Annis, J., Avila, S., Bhargava, S., Bridle, S.L., Brooks, D., Brout, D., Castander, F.J., Cawthon, R., Chang, C., Conselice, C., Costanzi, M., Crocce, M., Costa, L.N.d., Pereira, M.E.S., Davis, T.M., Desai, S., Dietrich, J.P., Doel, P., Eckert, K., Evrard, A.E., Ferrero, I., Fosalba, P., García-Bellido, J., Gerdes, D.W., Giannantonio, T., Gruen, D., Gutierrez, G., Hinton, S.R., Hollowood, D.L., Honscheid, K., Huff, E.M., Huterer, D., James, D.J., Jeltema, T., Kuehn, K., Lahav, O., Lidman, C., Lima, M., Lin, H., Maia, M.A.G., Marshall, J.L., Martini, P., Melchior, P., Miquel, R., Mohr, J.J., Morgan, R., Neilsen, E., Plazas, A.A., Romer, A.K., Roodman, A., Sanchez, E., Scarpine, V., Schubnell, M., Serrano, S., Smith, M., Suchyta, E., Tarle, G., Thomas, D., To, C., Varga, T.N., Wechsler, R.H., Weller, J., Wilkinson, R.D., 2021. Dark Energy Survey Year 3 Results: Photometric Data Set for Cosmology. ApJS 254, 24. URL: https://doi.org/10.3847/1538-4365/abeb66, doi:10.3847/1538-4365/abeb66. publisher: American Astronomical Society.

Simon, J.D., 2019. The Faintest Dwarf Galaxies. Annual Review of Astronomy and Astrophysics 57, 375–415. URL: https://ui.adsabs.harvard.edu/abs/2019ARA&A..57..375S/abstract, doi:10.1146/annurev-astro-091918-104453.

Stein, G., Blaum, J., Harrington, P., Medan, T., Lukic, Z., 2022. Mining for Strong Gravitational Lenses with Self-supervised Learning. The Astrophysical Journal 932, 107. URL: http://arxiv.org/abs/2110.00023, doi:10.3847/1538-4357/ac6d63. arXiv:2110.00023 [astro-ph].

Tanoglidis, D., Drlica-Wagner, A., Wei, K., Li, T.S., Sánchez, J., Zhang, Y., Peter, A.H.G., Feldmeier-Krause, A., Prat, J., Casey, K., Palmese, A., Sánchez, C., DeRose, J., Conselice, C., Gagnon, L., Abbott, T.M.C., Aguena, M., Allam, S., Avila, S., Bechtol, K., Bertin, E., Bhargava, S., Brooks, D., Burke, D.L., Rosell, A.C., Kind, M.C., Carretero, J., Chang, C., Costanzi, M., Costa, L.N.d., Vicente, J.D., Desai, S., Diehl, H.T., Doel, P., Eifler, T.F., Everett, S., Evrard, A.E., Flaugher, B., Frieman, J., García-Bellido, J., Gerdes, D.W., Gruendl, R.A., Gschwend, J., Gutierrez, G., Hartley, W.G., Hollowood, D.L., Huterer, D., James, D.J., Krause, E., Kuehn, K., Kuropatkin, N., Maia, M.A.G., March, M., Marshall, J.L., Menanteau, F., Miquel, R., Ogando, R.L.C., Paz-Chinchón, F., Romer, A.K., Roodman, A., Sanchez, E., Scarpine, V., Serrano, S., Sevilla-Noarbe, I., Smith, M., Suchyta, E., Tarle, G., Thomas, D., Tucker, D.L., Walker, A.R., DES Collaboration, 2021a. Shadows in the Dark: Low-surface-brightness Galaxies Discovered in the Dark Energy Survey. ApJS 252, 18. URL: https://iopscience.iop.org/article/10.3847/1538-4365/abca89, doi:10.3847/1538-4365/abca89.

Tanoglidis, D., Ćiprijanović, A., Drlica-Wagner, A., 2021b. Deepshadows: Separating low surface brightness galaxies from artifacts using deep learning. Astronomy and Computing 35, 100469. URL: https://www.sciencedirect.com/science/article/pii/S2213133721000238, doi:https://doi.org/10.1016/j.ascom.2021.100469.

Teeninga, P., Moschini, U., Trager, S.C., Wilkinson, M.H.F., 2015. Improved Detection of Faint Extended Astronomical Objects Through Statistical Attribute Filtering, in: Benediktsson, J.A., Chanussot, J., Najman, L., Talbot, H. (Eds.), Mathematical Morphology and Its Applications to Signal and Image Processing, Springer International Publishing, Cham. pp. 157–168. doi:10.1007/978-3-319-18720-4_14.

Valenzuela, L., Pichara, K., 2018. Unsupervised Classification of Variable Stars. Monthly Notices of the Royal Astronomical Society 474, 3259–3272. URL: http://arxiv.org/abs/1801.09723, doi:10.1093/mnras/stx2913. arXiv:1801.09723 [astro-ph].

Wang, J., Shen, H.T., Song, J., Ji, J., 2014. Hashing for Similarity Search: A Survey. URL: http://arxiv.org/abs/1408.2927. arXiv:1408.2927 [cs].

Yi, Z., Li, J., Du, W., Liu, M., Liang, Z., Xing, Y., Pan, J., Bu, Y., Kong, X., Wu, H., 2022. Automatic detection of low surface brightness galaxies from SDSS images. Monthly Notices of the Royal Astronomical Society 513, 3972–3981. URL: http://arxiv.org/abs/2203.16813, doi:10.1093/mnras/stac775. arXiv:2203.16813 [astro-ph].

Zaritsky, D., Donnerstein, R., Dey, A., Kadowaki, J., Zhang, H., Karunakaran, A., Martínez-Delgado, D., Rahman, M., Spekkens, K., 2018. Systematically Measuring Ultra-diffuse Galaxies (SMUDGes). I. Survey Description and First Results in the Coma Galaxy Cluster and Environs. ApJS 240, 1. URL: https://doi.org/10.3847/1538-4365/aaefe9, doi:10.3847/1538-4365/aaefe9. publisher: American Astronomical Society.

Zezula, P., Amato, G., Dohnal, V., Batko, M., 2006. Similarity Search - The Metric Space Approach. volume 32. doi:10.1007/0-387-29151-2.

Łukasik, S., Lalik, K., Sarna, P., Kowalski, P.A., Charytanowicz, M., Kulczycki, P., 2019. Efficient Astronomical Data Condensation Using Approximate Nearest Neighbors. International Journal of Applied Mathematics and Computer Science 29, 467–476. doi:10.2478/amcs-2019-0034.

6 Apêndice

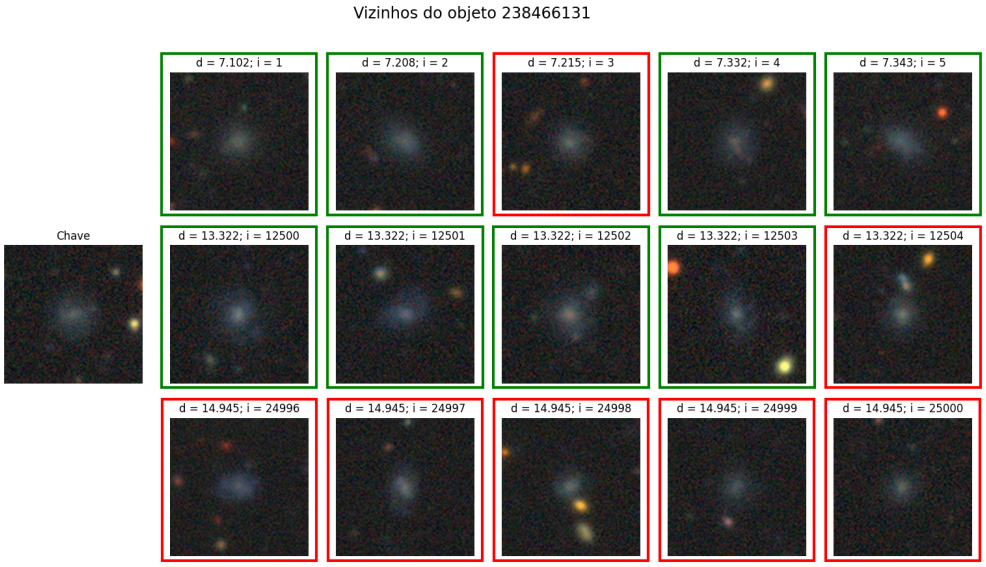


Figura 17: Vizinhos mais próximos da LSBG de `ID=238466131`. A estrutura da figura é igual à Figura 13.

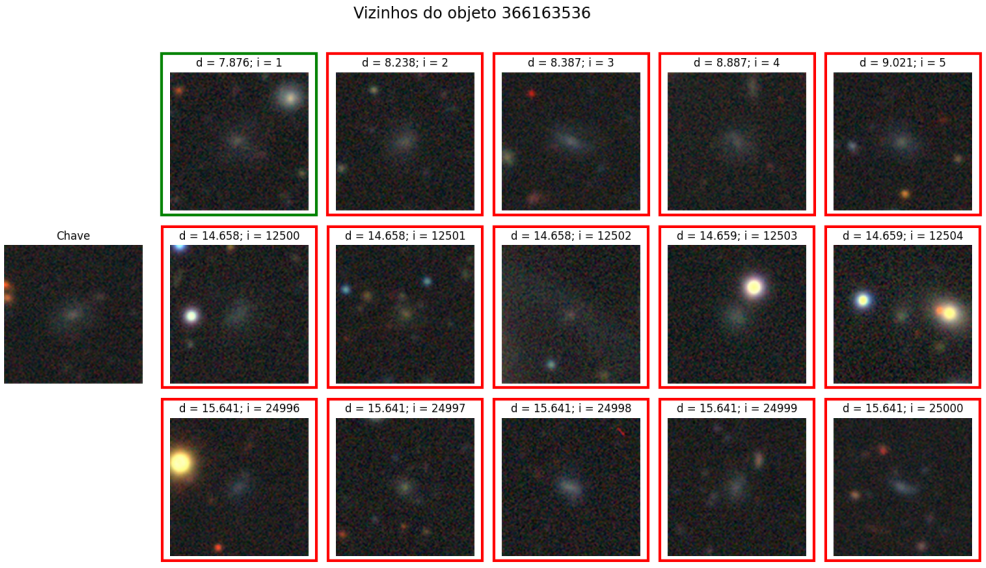


Figura 18: Vizinhos mais próximos da LSBG de `ID=366163536`. A estrutura da figura é igual à Figura 13.

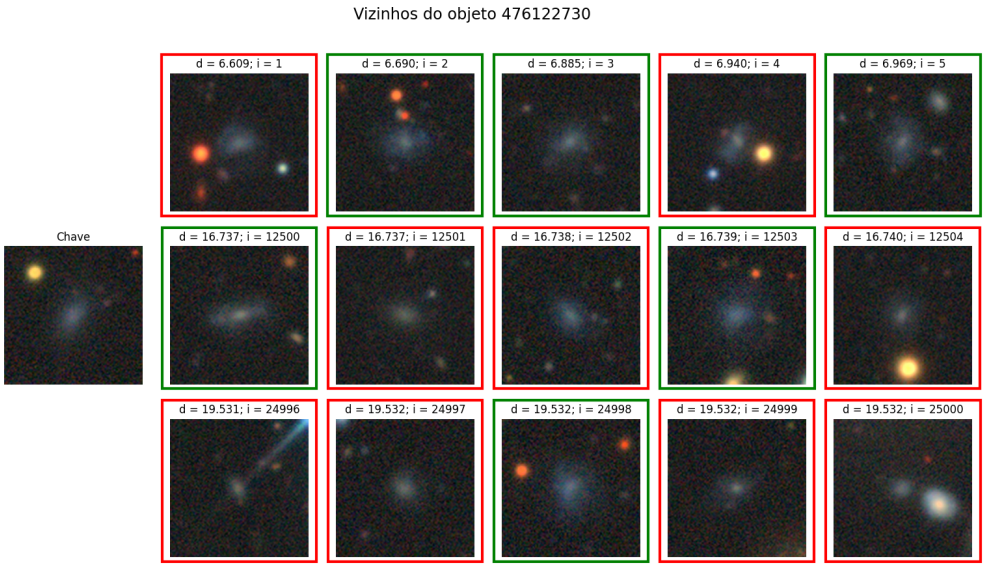


Figura 19: Vizinhos mais próximos da LSBG de `ID=476122730`. A estrutura da figura é igual à Figura 13.

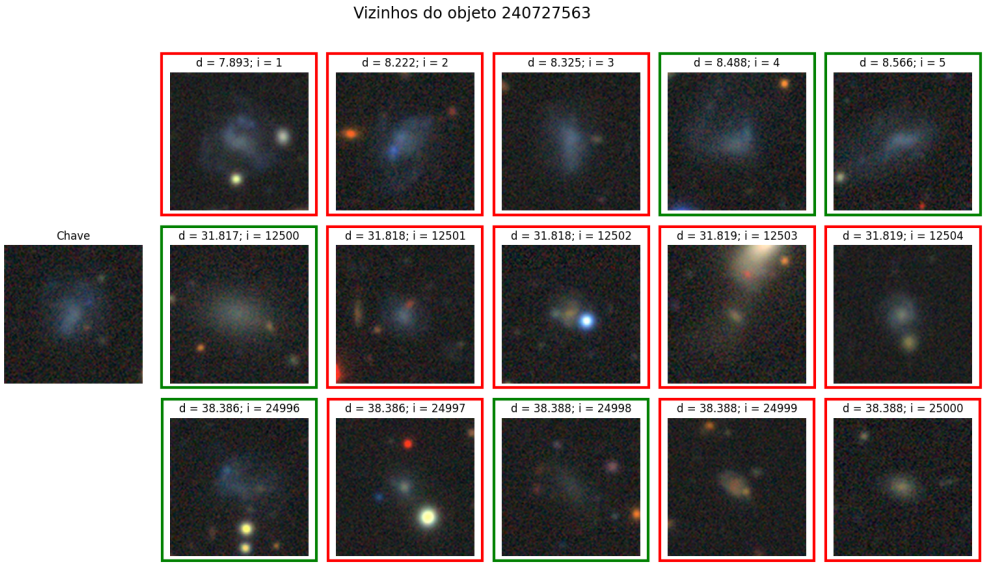


Figura 20: Vizinhos mais próximos da LSBG de `ID=240727563`. A estrutura da figura é igual à Figura 13.

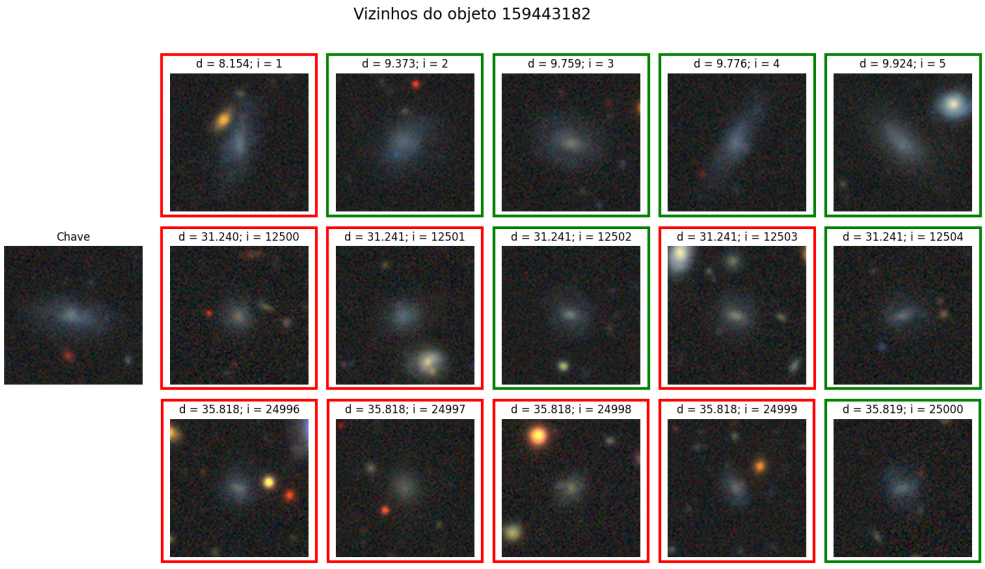


Figura 21: Vizinhos mais próximos da LSBG de `ID=159443182`. A estrutura da figura é igual à Figura 13.

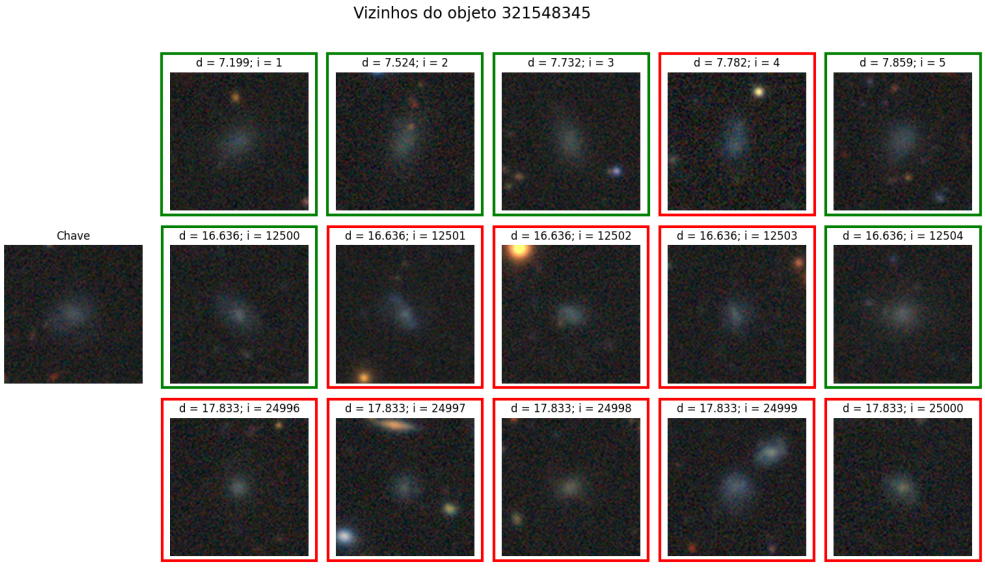


Figura 22: Vizinhos mais próximos da LSBG de `ID=321548345`. A estrutura da figura é igual à Figura 13.

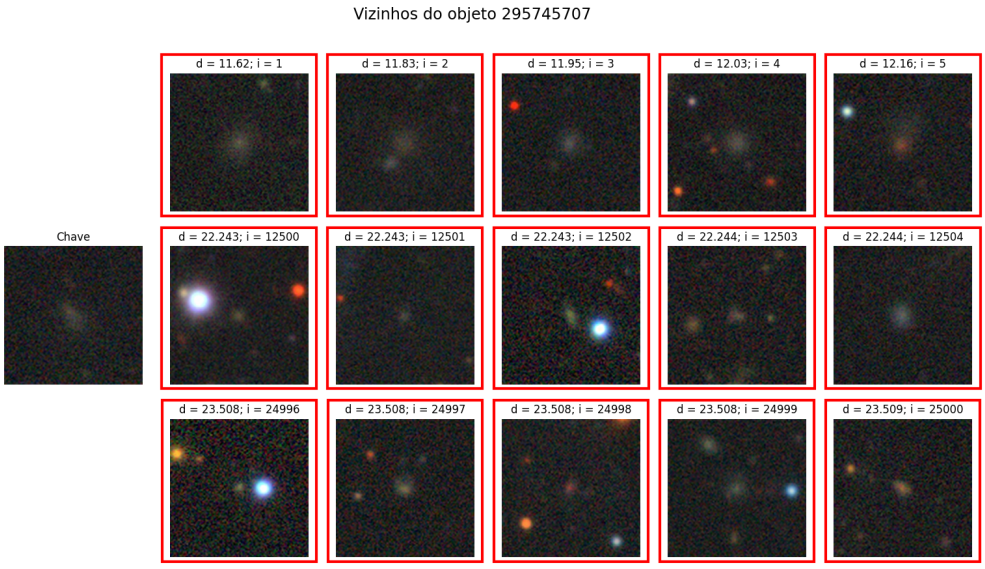


Figura 23: Vizinhos mais próximos da LSBG de `ID=295745707`. A estrutura da figura é igual à Figura 13.

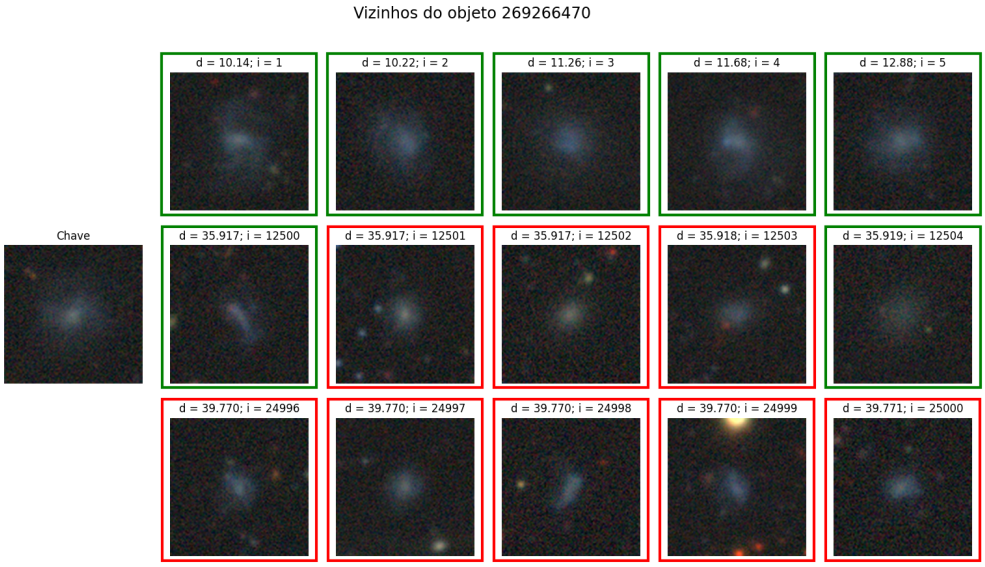


Figura 24: Vizinhos mais próximos da LSBG de `ID=269266470`. A estrutura da figura é igual à Figura 13.

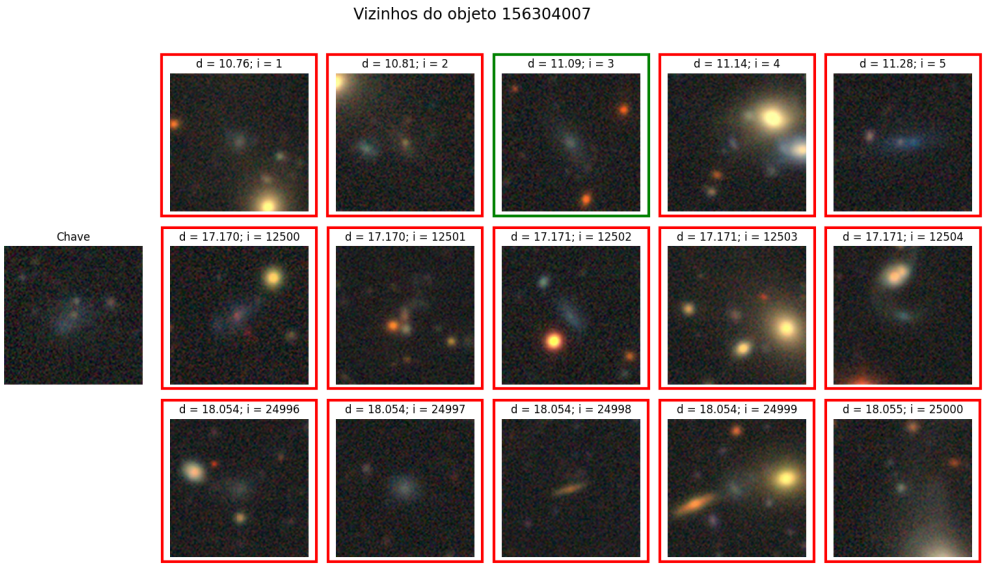


Figura 25: Vizinhos mais próximos da LSBG de `ID=156304007`. A estrutura da figura é igual à Figura 13.

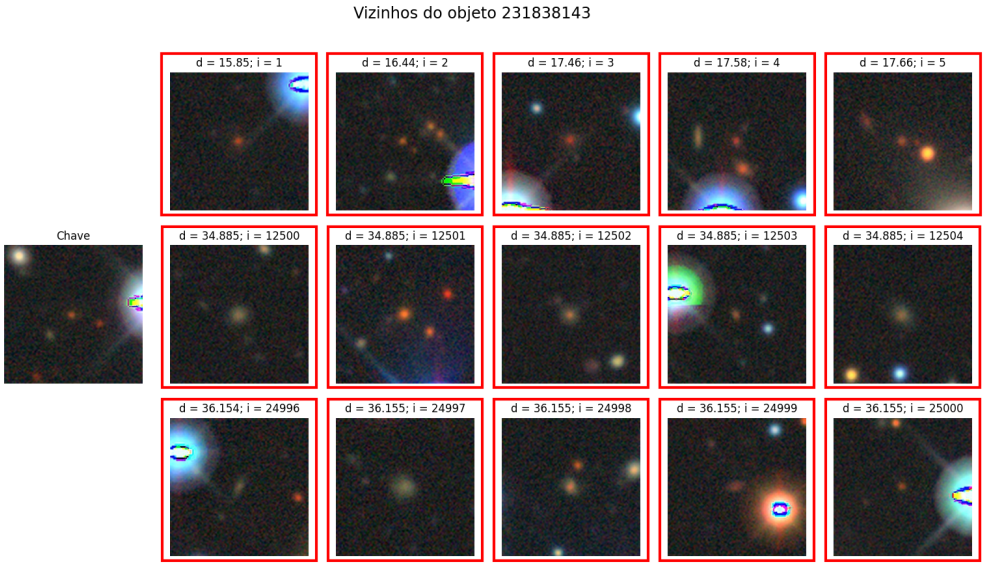


Figura 26: Vizinhos mais próximos do artefato tipo 1 de `ID=231838143`. A estrutura da figura é igual à Figura 13.

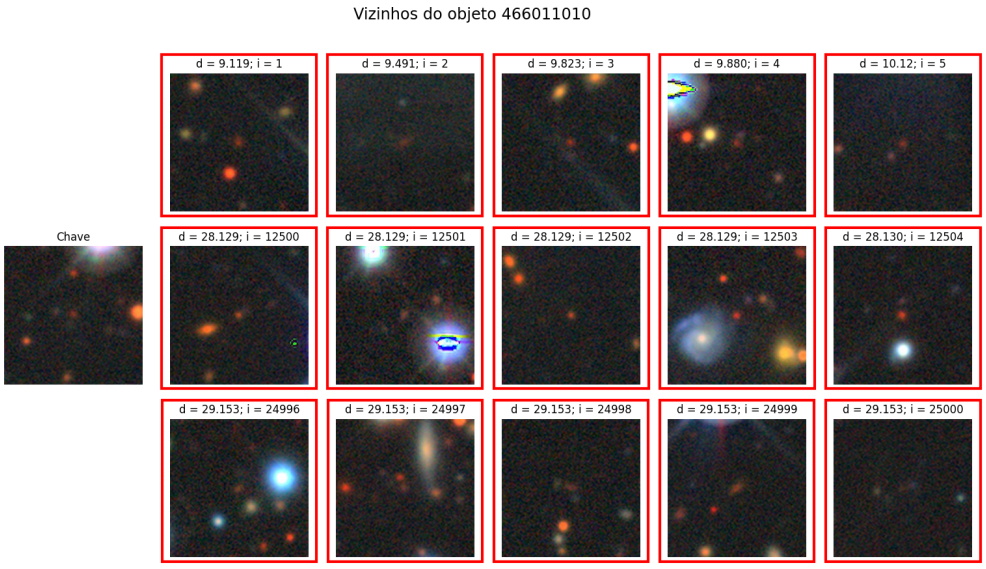


Figura 27: Vizinhos mais próximos do artefato tipo 1 de `ID=466011010`. A estrutura da figura é igual à Figura 13.

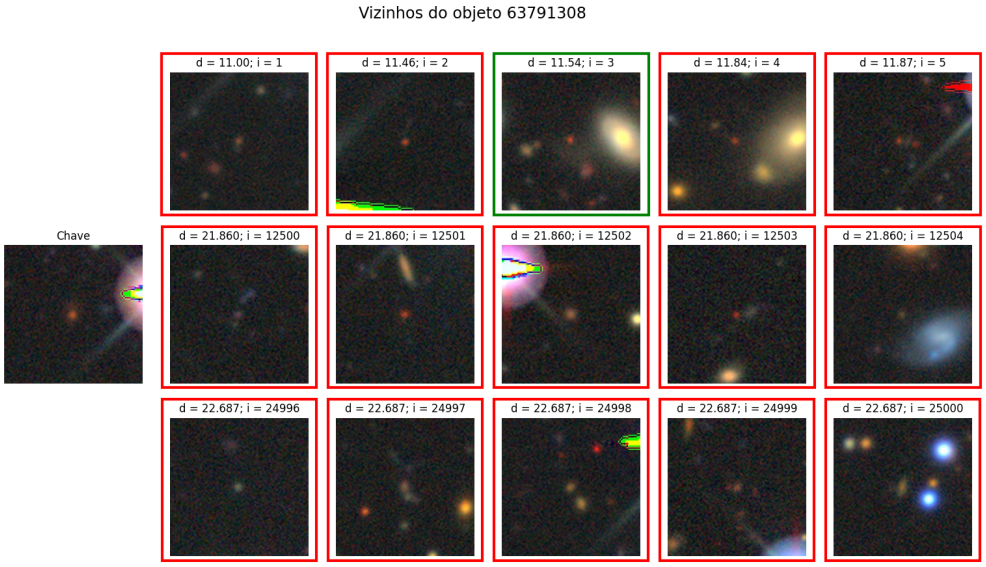


Figura 28: Vizinhos mais próximos do artefato tipo 1 de `ID=63791308`. A estrutura da figura é igual à Figura 13.

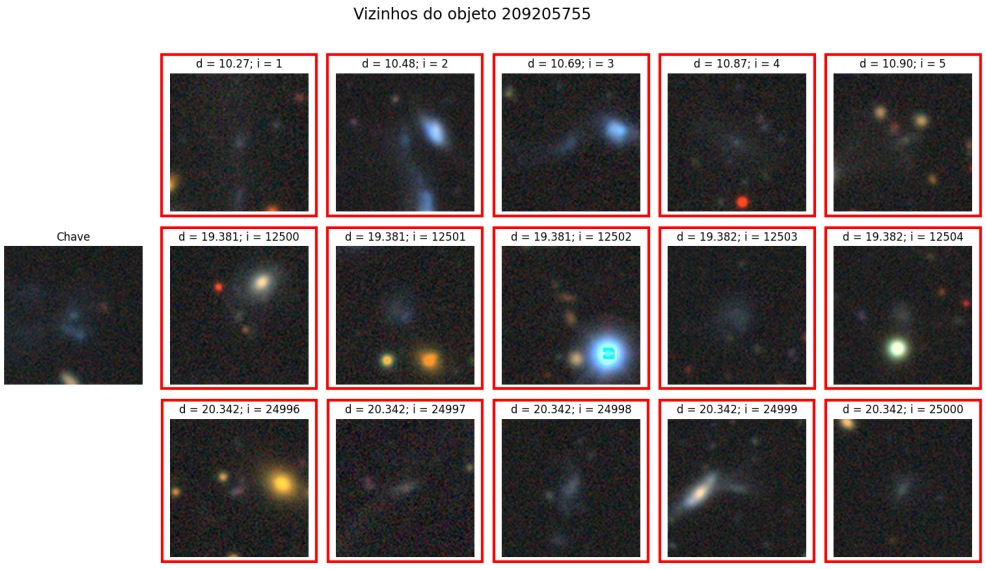


Figura 29: Vizinhos mais próximos do artefato tipo 1 de `ID=209205755`. A estrutura da figura é igual à Figura 13.

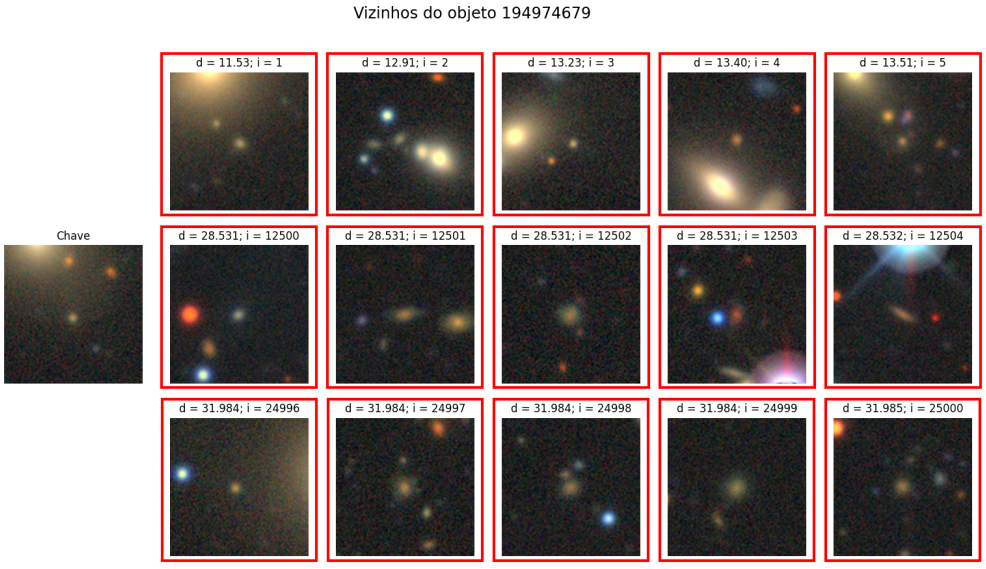


Figura 30: Vizinhos mais próximos do artefato tipo 1 de `ID=194974679`. A estrutura da figura é igual à Figura 13.

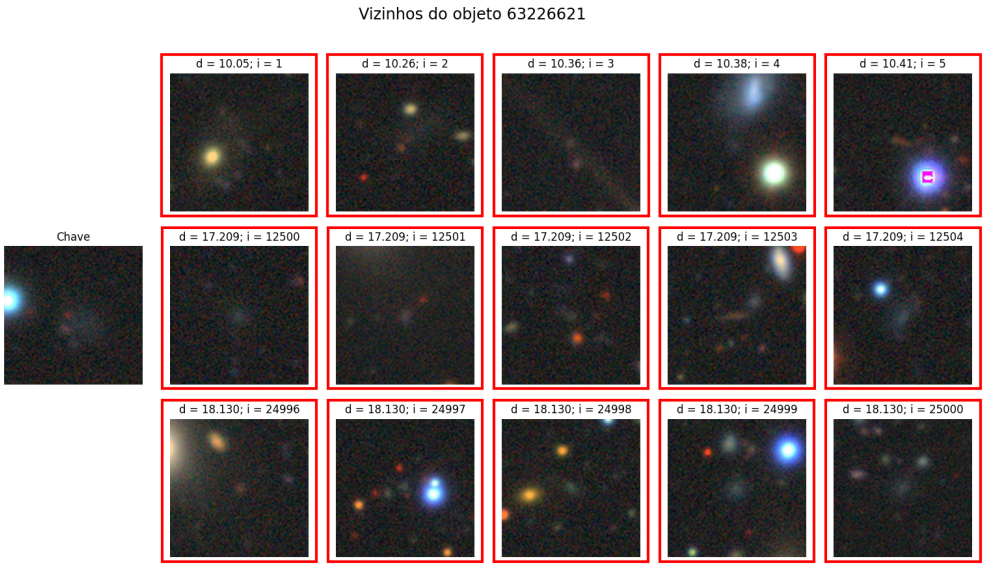


Figura 31: Vizinhos mais próximos do artefato tipo 2 de `ID=63226621`. A estrutura da figura é igual à Figura 13.

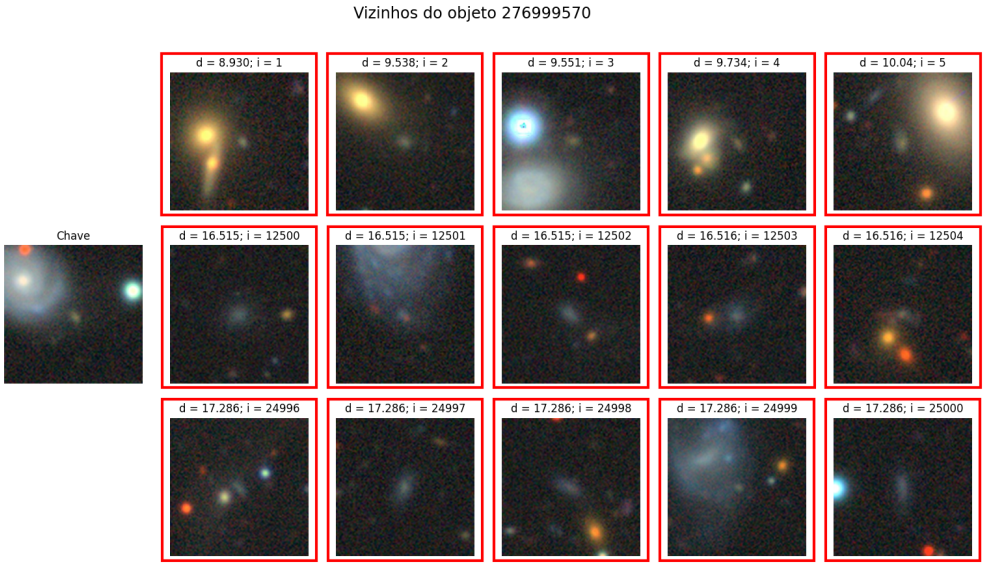


Figura 32: Vizinhos mais próximos do artefato tipo 2 de `ID=276999570`. A estrutura da figura é igual à Figura 13.

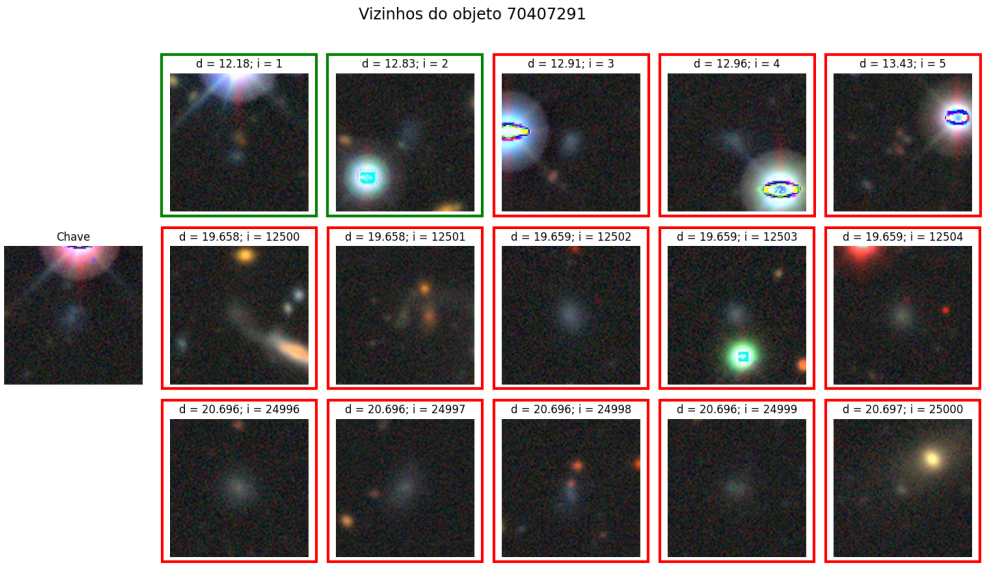


Figura 33: Vizinhos mais próximos do artefato tipo 2 de `ID=70407291`. A estrutura da figura é igual à Figura 13.

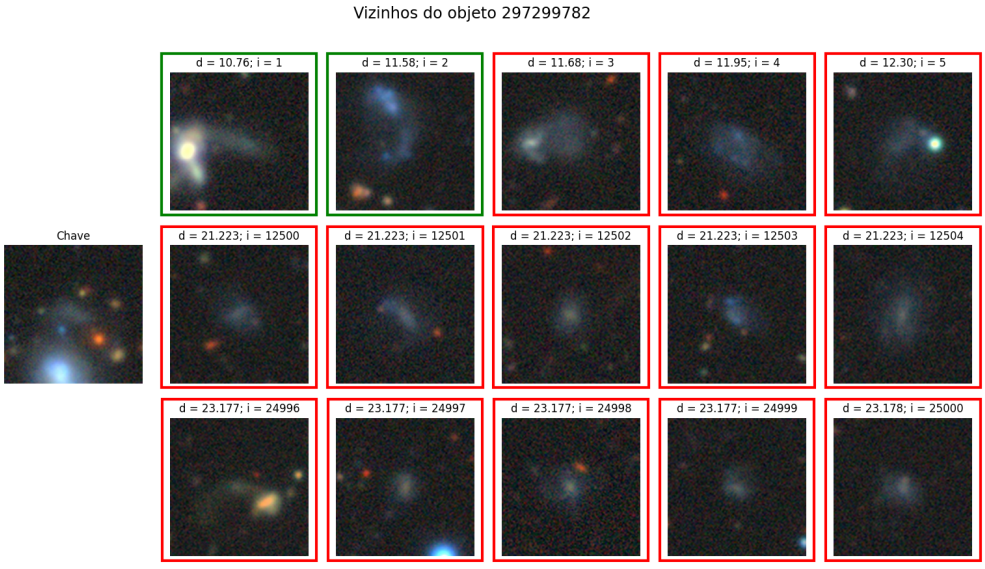


Figura 34: Vizinhos mais próximos do artefato tipo 2 de `ID=297299782`. A estrutura da figura é igual à Figura 13.