

DESENVOLVIMENTO DE UM PIPELINE EM PYTHON PARA DETERMINAÇÃO AUTOMÁTICA DA IDADE E DISTÂNCIA DE AGLOMERADOS ESTELARES A PARTIR DE DADOS FOTOMÉTRICOS DO SLOAN DIGITAL SKY SURVEY

DEVELOPMENT OF A PYTHON PIPELINE FOR AUTOMATIC DETERMINATION OF THE AGE AND DISTANCE OF STAR CLUSTERS FROM PHOTOMETRIC DATA FROM THE SLOAN DIGITAL SKY SURVEY

Guilherme Albuquerque Bellini

UNESP Universidade Estadual Paulista "Júlio de Mesquita Filho", Faculdade de Ciências,
Departamento de Física e Meteorologia, Curso de Física, Observatório Didático de Astronomia da
UNESP Bauru, guilherme.bellini@unesp.br

Resumo: *Desenvolveu-se um pipeline automatizado em Python para a determinação da distância e da idade de aglomerados estelares a partir de dados fotométricos do Sloan Digital Sky Survey (SDSS). O algoritmo realiza, de forma integrada, a filtragem dos dados, a construção do diagrama de cor-magnitude (CMD), a extração da sequência principal (MS), a identificação automática do ponto de turn-off e a leitura de isócronas do Dartmouth Stellar Evolution Database. O módulo de distância é estimado por meio de uma rotina de pareamento cor-magnitude, que calcula $m - M$ por minimização estatística. Aplicado ao aglomerado NGC 2266, o pipeline obteve $\mu = 14,43$ (distância de 7,7 kpc) e melhor compatibilidade com uma isócrona de 1,2 Gyr para metalicidade solar. Os resultados concordam com ajustes manuais, validando a abordagem.*

Palavras-chave: Aglomerados estelares; isócronas; Python; SDSS; evolução estelar.

Abstract: *An automated pipeline was developed in Python to determine the distance and age of star clusters from photometric data from the Sloan Digital Sky Survey (SDSS). The algorithm performs, in an integrated manner, data filtering, construction of the color-magnitude diagram (CMD), main sequence (MS) extraction, automatic identification of the turn-off point, and reading of isochrones from the Dartmouth Stellar Evolution Database. The distance modulus is estimated using a color-magnitude matching routine, which calculates $m - M$ by statistical minimization. Applied to the cluster NGC 2266, the pipeline obtained $\mu = 14.43$ (distance of 7.7 kpc) and better compatibility with a 1.2 Gyr isochrone for solar metallicity. The results agree with manual adjustments, validating the approach.*

Keywords: Star clusters; isochrones; Python; SDSS; stellar evolution.

INTRODUÇÃO

A determinação de parâmetros físicos de aglomerados estelares, como idade, distância e composição química, é fundamental para a compreensão da evolução estelar e da estrutura da nossa galáxia. Esses sistemas reúnem estrelas formadas aproximadamente na mesma época e com composição semelhante, o que os torna excelentes laboratórios naturais para testar modelos teóricos de evolução estelar. Entre as principais ferramentas utilizadas nesses estudos está o diagrama de



cor-magnitude (CMD), que permite comparar observações fotométricas com previsões teóricas expressas por isócronas, curvas que representam estrelas de mesma idade com diferentes massas e estágios evolutivos.

Em uma etapa inicial desta pesquisa, foi construído o CMD do aglomerado aberto NGC 2266 a partir de dados do Sloan Digital Sky Survey (SDSS) e ajustada uma isócrona teórica do Dartmouth Stellar Evolution Database. Por meio desse procedimento manual, foi possível estimar o módulo de distância e a idade aproximada do aglomerado, aplicando fundamentos de fotometria, modelos estelares e interpretação de dados astronômicos.

A partir dessa experiência prática, surgiu o interesse em automatizar o processo de análise de aglomerados estelares. A ideia central é implementar um pipeline em linguagem Python capaz de realizar automaticamente as etapas de leitura dos dados observacionais, construção do CMD, ajuste do módulo de distância e comparação entre isócronas de diferentes idades e metalicidades. Essa abordagem computacional visa reduzir a subjetividade do ajuste visual e aumentar a eficiência e a reprodutibilidade das análises. Além disso, o desenvolvimento dessa ferramenta tem potencial para aplicações didáticas e científicas, permitindo que estudantes e pesquisadores possam reproduzir e expandir seus estudos de caracterização estelar.

O presente trabalho tem como objetivo principal desenvolver e testar um pipeline automatizado em Python para a determinação da idade e da distância de aglomerados estelares a partir de dados fotométricos do SDSS. Especificamente, busca-se automatizar o ajuste do módulo de distância, implementar a comparação entre diferentes isócronas e avaliar o desempenho da ferramenta utilizando o aglomerado NGC 2266 como caso de estudo.

FUNDAMENTAÇÃO TEÓRICA

O estudo de aglomerados estelares é uma das principais ferramentas observacionais para compreender a formação e a evolução das estrelas na Via Láctea. Esses sistemas, por reunirem estrelas formadas simultaneamente a partir de uma mesma nuvem molecular, apresentam composição química semelhante e idades próximas. Isso os torna excelentes laboratórios naturais para testar modelos teóricos de evolução estelar (Binney; Merrifield, 1998).

A análise de aglomerados é tradicionalmente realizada por meio do diagrama de cor-magnitude (CMD), também conhecido como diagrama de Hertzsprung–Russell observacional. Nele, as magnitudes aparentes das estrelas são plotadas em função de seus índices de cor, permitindo identificar padrões como a sequência principal, o ponto de turn-off e o ramo das gigantes vermelhas. O formato do CMD fornece informações diretas sobre a idade e a composição do aglomerado (Salaris; Cassisi, 2005).

O uso de isócronas teóricas é essencial nesse tipo de análise. As isócronas representam curvas que conectam pontos de igual idade em diagramas teóricos de magnitude e temperatura efetiva. A comparação entre essas curvas e os dados observacionais permite estimar parâmetros como idade, distância e metalicidade. O Dartmouth Stellar Evolution Database, desenvolvido por Dotter et al. (2008), é amplamente utilizado por fornecer modelos de isócronas com diferentes idades e

metallicidades, compatíveis com diversos sistemas fotométricos, incluindo o SDSS ugriz, utilizado neste trabalho.

A base de dados fotométricos do Sloan Digital Sky Survey (SDSS) revolucionou a Astronomia observacional ao disponibilizar medições precisas em larga escala. O SDSS fornece magnitudes em cinco filtros (u, g, r, i, z) e é amplamente utilizado em estudos de populações estelares e galácticas (York et al., 2000). A qualidade e a abrangência de seus dados permitem a construção de CMDs detalhados e a comparação direta com modelos teóricos.

Nos últimos anos, o avanço das ferramentas computacionais possibilitou a aplicação de métodos automáticos e estatísticos para o ajuste de isócronas. Técnicas como minimização de erro quadrático e ajuste por X^2 têm sido utilizadas para identificar, de forma objetiva, a isócrona que melhor se ajusta às observações (Monteiro, Dias, Santiago, 2021; Dias et al., 2012).

Essa abordagem reduz a subjetividade do ajuste visual e favorece a reprodutibilidade dos resultados, o que se alinha ao objetivo deste trabalho: desenvolver um pipeline automatizado em Python que realize esse processo de maneira integrada e sistemática.

METODOLOGIA

Esta pesquisa possui caráter exploratório e computacional, com abordagem quantitativa e aplicada, uma vez que busca desenvolver e testar uma ferramenta automatizada para análise de dados astronômicos. O estudo tem como base o uso de dados fotométricos públicos e modelos teóricos consolidados na literatura, implementando um pipeline em linguagem Python para automatizar procedimentos tradicionalmente realizados de forma manual.

Dados fotométricos e pré-processamento

Os dados fotométricos do SDSS foram obtidos a partir de uma planilha contendo as magnitudes corrigidas g_0 e r_0 para as estrelas na direção do aglomerado NGC 2266. O pipeline inicia com a leitura dessa tabela utilizando a biblioteca Pandas, seguida de verificações automáticas da presença das colunas necessárias e da remoção de registros com valores ausentes ou incertezas fotométricas elevadas. Em seguida, é calculado o índice de cor $g-r$, utilizado como eixo horizontal no CMD, enquanto r_0 compõe o eixo vertical.

O pipeline inicia com a leitura da tabela fotométrica do SDSS contendo as magnitudes corrigidas g_0 e r_0 do aglomerado NGC 2266. A listagem 1 mostra a função responsável por carregar o arquivo, verificar a presença das colunas necessárias e calcular o índice de cor $g-r$.

```
1 import pandas as pd
2
3 def load_sdss_table(filepath):
4     """Carrega a tabela fotométrica e calcula a cor (g-r)."""
```

```

5     df = pd.read_excel(filepath)                # ou read_csv, conforme o
        arquivo
6
7     required = ["g0", "r0"]
8     for col in required:
9         if col not in df.columns:
10             raise ValueError(f"Coluna obrigat ria ausente: {col}")
11
12     # Remove linhas com valores ausentes
13     df = df.dropna(subset=required)
14
15     # Calcula o ndice de cor
16     df["g_r"] = df["g0"] - df["r0"]
17
18     # Filtro simples de qualidade em magnitude
19     df = df[df["r0"] < 20]
20
21     return df

```

Figura 01: Listagem 1 - Leitura e pré-processamento dos dados fotométricos do SDSS

Os dados são carregados com a biblioteca Pandas, que permite manipulação eficiente de tabelas. Primeiramente, a função verifica se as colunas obrigatórias (g0 e r0) estão presentes; em seguida, remove linhas com valores ausentes e calcula o índice de cor g-r, armazenado na “coluna g_r” da programação. Por fim, aplica-se um corte em magnitude aparente para reduzir o impacto de estrelas muito fracas, onde os erros fotométricos são maiores.

Estrutura do pipeline em Python

O código foi organizado em módulos lógicos correspondentes às principais tarefas do pipeline:

- leitura e filtragem dos dados fotométricos;
- construção do CMD observacional;
- determinação automática da sequência principal (MS) e do ponto de turn-off ;
- classificação preliminar das estrelas em MS e ramo das gigantes vermelhas (RGB);
- leitura e interpretação das isócronas do Dartmouth Stellar Evolution Database;
- ajuste do módulo de distância μ por comparação entre dados observacionais e isócronas;
- geração dos diagramas finais (CMD com isócrona deslocada e diagrama HR em magnitudes absolutas).



As bibliotecas principais utilizadas foram Pandas e NumPy para manipulação de dados, Matplotlib para visualização gráfica e rotinas de interpolação e ajuste de curvas da biblioteca SciPy. O código completo é modular, permitindo a reaplicação das mesmas funções a outros aglomerados estelares.

Extração da sequência principal e detecção do turn-off

Para a identificação da sequência principal (Main Sequence, MS), o conjunto de dados é segmentado em intervalos regulares de cor g-r; para cada bin é calculada a mediana da magnitude r0. A sequência de medianas é então suavizada por meio de uma janela móvel e ajustada por uma spline cúbica, resultando em uma curva contínua que representa a crista da MS observada.

A partir dessa spline, o código calcula a derivada primeira e a derivada segunda com respeito à cor. O ponto de maior curvatura na parte brilhante da sequência — isto é, aquele em que a derivada segunda assume módulo máximo — é interpretado como o ponto de turn-off. Esse procedimento é inteiramente automático e independe de escolha visual por parte do usuário.

Uma classificação preliminar das estrelas em MS e RGB é feita comparando a magnitude observada com a magnitude prevista pela spline da MS. Estrelas significativamente mais brilhantes e mais vermelhas do que a MS são marcadas como candidatas a gigantes vermelhas, permitindo isolar a evolução pós-sequência principal no CMD.

A sequência principal é obtida de forma automática a partir da distribuição de pontos no CMD. A listagem 2 apresenta o trecho de código responsável por construir a crista da MS, ajustando uma spline cúbica e identificando o ponto de maior curvatura, associado ao turn-off point.

```
--
1 import numpy as np
2 from scipy.interpolate import make_interp_spline
3
4 def main_sequence_spline(df, bin_width=0.05):
5     # Binagem em cor
6     bins = np.arange(df["g_r"].min(), df["g_r"].max() + bin_width,
7                       bin_width)
8     centers, med_r = [], []
9
10    for i in range(len(bins) - 1):
11        in_bin = df[(df["g_r"] >= bins[i]) & (df["g_r"] < bins[i
12                        +1])]
13
```

```

11         if len(in_bin) > 5:
12             centers.append(0.5 * (bins[i] + bins[i+1]))
13             med_r.append(in_bin["r0"].median())
14
15     centers = np.array(centers)
16     med_r = np.array(med_r)
17
18     # Ajuste de spline c bica para suavizar a MS
19     spline = make_interp_spline(centers, med_r, k=3)
20     c_smooth = np.linspace(centers.min(), centers.max(), 300)
21     r_smooth = spline(c_smooth)
22
23     # Derivadas para localizar o turn-off
24     dr_dc = np.gradient(r_smooth, c_smooth)
25     d2r_dc = np.gradient(dr_dc, c_smooth)
26
27     # Ponto de maior curvatura na região mais brilhante
28     idx_to = np.argmin(d2r_dc)
29     color_to = c_smooth[idx_to]
30     r0_to = r_smooth[idx_to]
31
32     return c_smooth, r_smooth, color_to, r0_to

```

Figura 02: Listagem 2 - Construção da sequência principal e detecção automática do turn-off.

Inicialmente, o código divide o intervalo de cor em bins de largura fixa e calcula, em cada bin, a mediana da magnitude r_0 . Esses pontos medianos representam a crista da sequência principal observada. Em seguida, é ajustada uma spline cúbica, que produz uma curva suave $r(c)$ ao longo da cor.

As derivadas primeira e segunda dessa curva são obtidas numericamente com `np.gradient`. O ponto de maior curvatura na região mais brilhante da sequência principal é interpretado como o turn-off, pois marca a mudança de inclinação típica entre a MS e as fases evolutivas posteriores. A função retorna tanto a curva suavizada quanto as coordenadas (cor, r_0) do turn-off.

Leitura das isócronas e ajuste do módulo de distância

O pipeline realiza a leitura automatizada de isócronas do Dartmouth Stellar Evolution Database a partir de arquivos texto. O código identifica dinamicamente quais colunas correspondem às magnitudes teóricas nos filtros g e r , permitindo o uso de diferentes formatos de saída. Para cada ponto da isócrona, são calculados a cor $g-r$ e a magnitude absoluta M_r .

O ajuste do módulo de distância μ é realizado por meio de uma rotina de pareamento cor–magnitude: para cada intervalo de cor, o pipeline associa estrelas

observadas e pontos da isócrona com cores semelhantes e calcula os valores de $m - M = r_0 - M_r$. A mediana dos valores válidos de $m - M$ é adotada como estimativa robusta do módulo de distância do aglomerado, minimizando a influência de *outliers*.

Uma vez determinado μ , o código gera dois diagramas principais: (i) o CMD observado com a isócrona deslocada para magnitudes aparentes ($M_r + \mu$); e (ii) o diagrama HR, em que as magnitudes absolutas observadas ($r_0 - \mu$) são comparadas diretamente com os valores teóricos da isócrona.

As isócronas do Dartmouth Stellar Evolution Database são lidas a partir de arquivos texto, e o módulo de distância é estimado por comparação cor–magnitude. A listagem 3 mostra o procedimento adotado.

```

1 def load_dartmouth_iso(filepath):
2     """L  arquivo de is crona Dartmouth e retorna cor (g-r) e Mr.
3         """
4
5     iso = pd.read_csv(filepath, comment="#", delim_whitespace=True)
6
7     # Exemplos de poss veis nomes de coluna
8     g_cols = [c for c in iso.columns if c.lower().startswith("g")]
9     r_cols = [c for c in iso.columns if c.lower().startswith("r")]
10
11     g = iso[g_cols[0]]
12     r = iso[r_cols[0]]
13
14     iso["g_r"] = g - r      # cor te rica
15     iso["Mr"] = r          # magnitude absoluta em r
16
17     return iso
18
19
20 def distance_modulus(df, iso, color_tol=0.02):
21     diffs = []
22
23     for _, row in iso.iterrows():
24         c_iso = row["g_r"]
25         Mr    = row["Mr"]
26
27         # estrelas observadas com cor pr xima da cor da is crona
28         mask = np.abs(df["g_r"] - c_iso) < color_tol
29         subset = df[mask]
30
31         if len(subset) > 0:

```

```

29         m_med = subset["r0"].median()
30         diffs.append(m_med - Mr)
31
32     mu = np.median(diffs)
33     d_pc = 10**((mu + 5)/5.0)
34
35     return mu, d_pc

```

Figura 03: Listagem 3 - Leitura da isócrona Dartmouth e cálculo do módulo de distância μ .

A função `load_dartmouth_iso` identifica automaticamente quais colunas do arquivo correspondem às magnitudes teóricas nos filtros *g* e *r*, calculando a cor *g-r* e a magnitude absoluta *Mr* para cada ponto da isócrona. Em seguida, a função `distance_modulus` realiza o pareamento cor–magnitude: para cada cor da isócrona, selecionam-se estrelas observadas com cor semelhante e calcula-se $m - M = r_0 - Mr$. A mediana desses valores define o módulo de distância μ , a partir do qual a distância *d* em parsecs é obtida via:

$$d = 10^{(\mu+5)/5}$$

Esse procedimento é robusto a *outliers* e permite ajustar automaticamente a isócrona aos dados observacionais, gerando, na etapa seguinte do pipeline, o CMD e o diagrama HR com a isócrona deslocada.

Descrição algorítmica do pipeline

O fluxo global do pipeline pode ser resumido no Algoritmo 1, que sintetiza as etapas implementadas em Python para o aglomerado NGC 2266.

Algorithm 1: Pipeline para determinação automática de distância e idade de um aglomerado estelar

Entrada: Tabela fotométrica com g_0 , r_0 e erros associados; arquivo de isócrona Dartmouth.

Saída : Módulo de distância μ , distância d , idade e diagramas CMD/HR com isócrona ajustada.

- 1 **1. Leitura e pré-processamento dos dados observacionais**
- 2 Ler tabela fotométrica (*Pandas*); remover linhas com valores ausentes
- 3 Aplicar cortes em erro fotométrico e magnitude; calcular cor ($g - r$)
- 4 **2. Construção do CMD e extração da sequência principal**
- 5 Dividir o intervalo de cor em bins; para cada bin, calcular mediana de r_0
- 6 Suavizar a sequência de medianas e ajustar spline cúbica à MS
- 7 Calcular derivadas da spline e localizar o máximo de curvatura (*turn-off*)
- 8 Classificar estrelas em MS/RGB a partir do desvio em relação à MS
- 9 **3. Leitura e preparação da isócrona teórica**
- 10 Ler arquivo de isócrona; identificar colunas g e r ; calcular $(g - r)$ e M_r
- 11 **4. Ajuste do módulo de distância**
- 12 Para cada faixa de cor, emparelhar estrelas observadas e pontos da isócrona
- 13 Calcular $m - M = r_0 - M_r$ e obter μ como mediana dos valores válidos
- 14 Calcular a distância $d = 10^{(\mu+5)/5}$ (pc)
- 15 **5. Geração dos diagramas finais**
- 16 Construir CMD com pontos observados e isócrona deslocada ($M_r + \mu$)
- 17 Construir diagrama HR com magnitudes absolutas ($r_0 - \mu$) e M_r
- 18 Salvar gráficos e parâmetros derivados (idade, μ , d)

Quadro 01: Algoritmo 1 - Fluxo global do pipeline.



ANÁLISE DE DADOS

A aplicação do pipeline ao aglomerado NGC 2266 foi realizada a partir da planilha fotométrica contendo as magnitudes corrigidas g_0 e r_0 e o índice de cor $g-r$. Após os cortes de qualidade em magnitude e incerteza fotométrica, o código plota o diagrama cor–magnitude (CMD) observacional e grava a imagem (Figura 4).

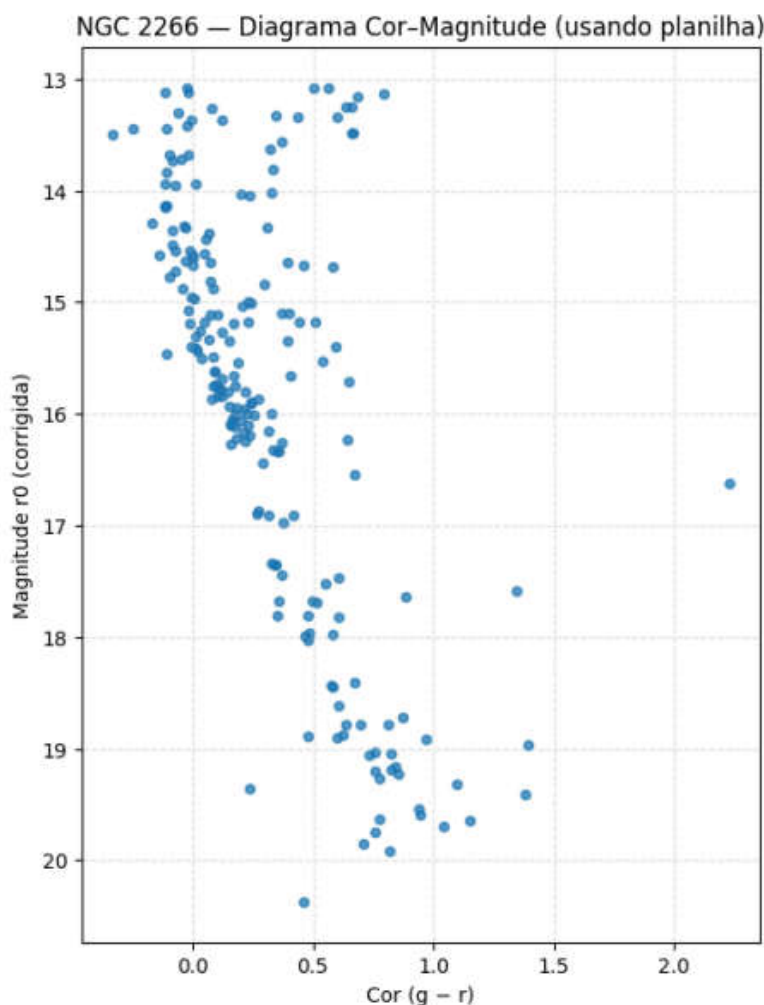


Figura 04: CMD do aglomerado NGC 2266 a partir dos dados fotométricos do SDSS.

Em seguida, o algoritmo constrói automaticamente a crista da sequência principal, calculando a mediana de r_0 em bins de cor e ajustando uma spline cúbica. A partir dessa curva suave, são computadas as derivadas primeira e segunda, permitindo localizar o ponto de maior curvatura associado ao turn-off. Ao mesmo tempo, o código interpola a posição da MS para cada estrela e classifica como candidatas a gigantes vermelhas aquelas que se encontram significativamente mais brilhantes e vermelhas do que a sequência principal. O resultado dessa classificação é apresentado na Figura 5.

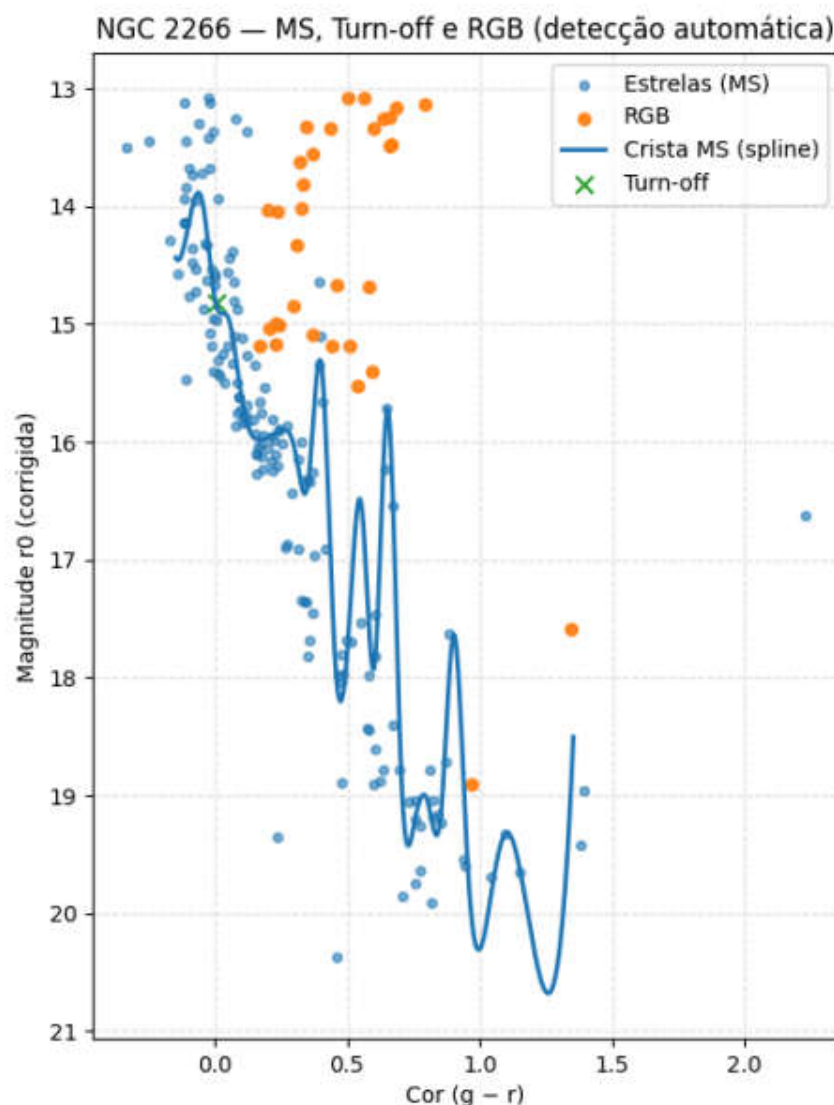


Figura 05: CMD com a sequência principal suavizada (spline), o ponto de turn-off e as estrelas classificadas como candidatas a RGB.

A etapa seguinte consiste na leitura da isócrona do Dartmouth Stellar Evolution Database e na estimativa do módulo de distância μ pelo pareamento cor–magnitude. Para cada ponto da isócrona, o código seleciona estrelas observadas com cor semelhante e calcula $m - M = r_0 - M_r$; a mediana desse conjunto é adotada como estimativa de μ . Com o valor encontrado, a isócrona é deslocada para magnitudes aparentes e sobreposta ao CMD, resultando na Figura 6.

Por fim, o módulo de distância é utilizado para converter as magnitudes aparentes em magnitudes absolutas, produzindo o diagrama de Hertzsprung–Russell (HR) observado. A comparação direta entre os dados em

magnitudes absolutas e a isócrona teórica é mostrada na Figura 7, que evidencia a boa concordância ao longo da sequência principal e do turn-off.

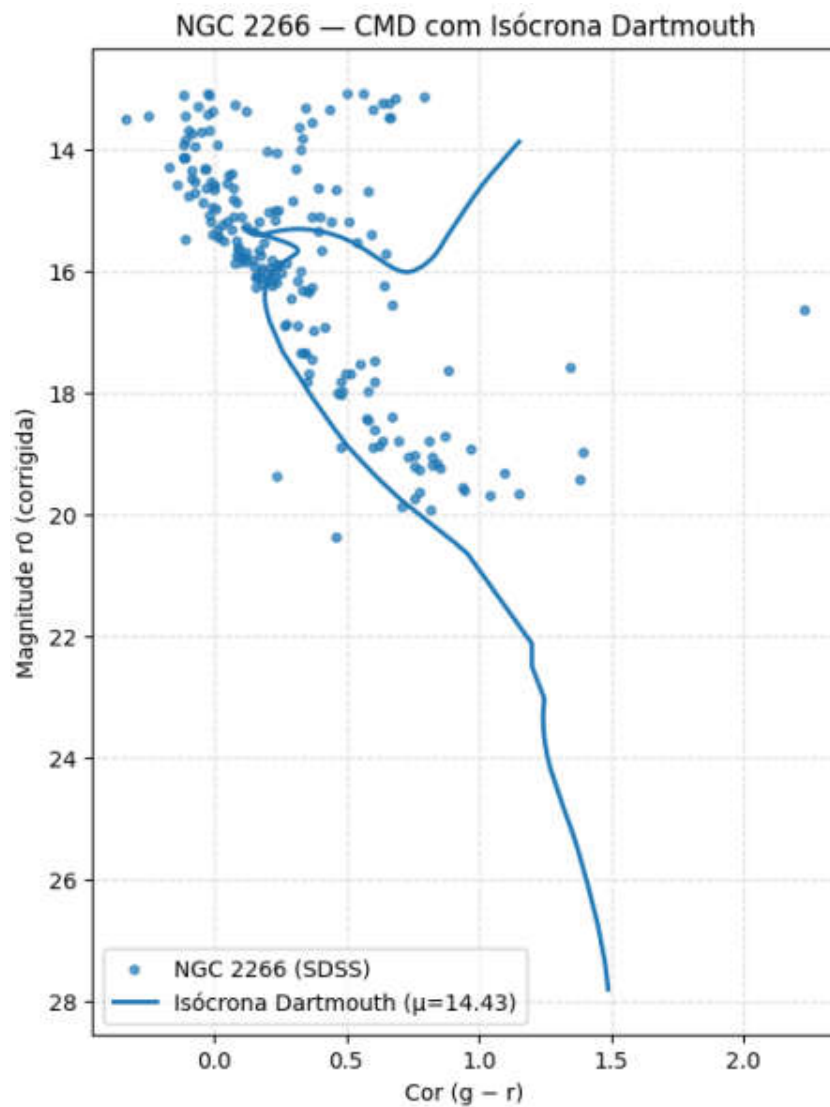


Figura 06: CMD do NGC 2266 com isócrona Dartmouth deslocada pelo módulo de distância estimado.

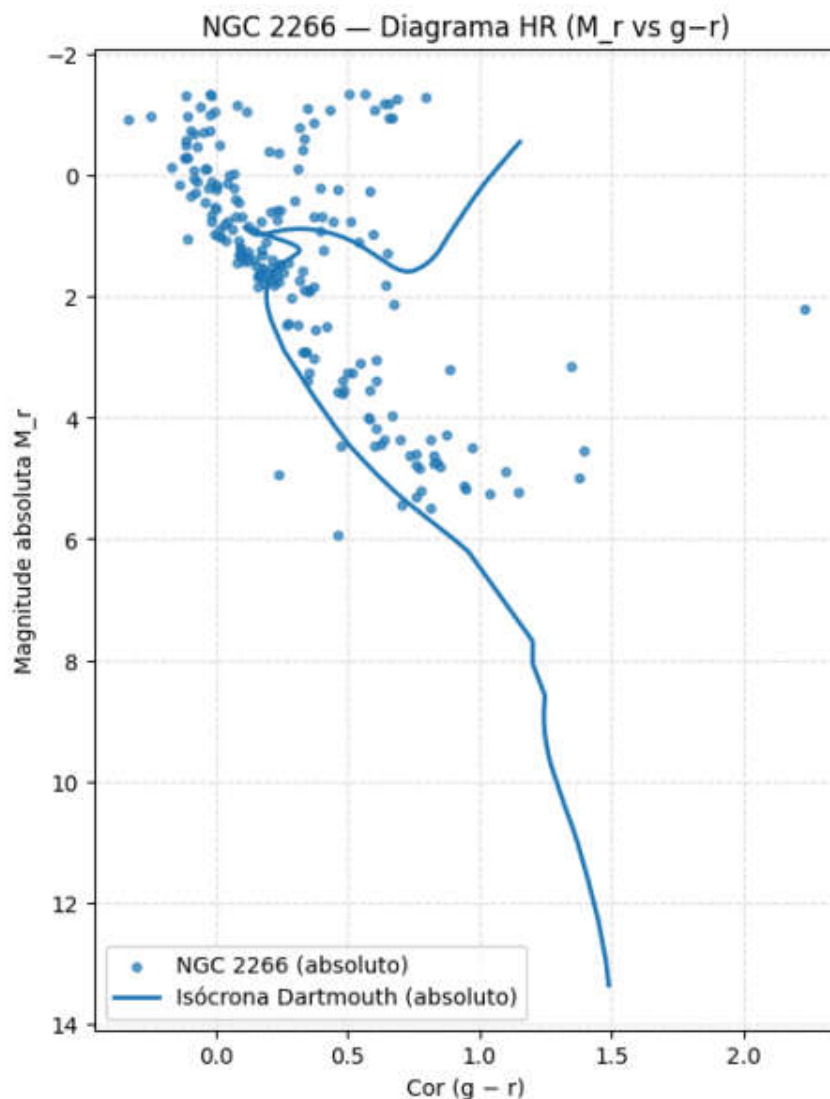


Figura 07: Diagrama HR do NGC 2266 em magnitudes absolutas, comparado à isócrona Dartmouth.

RESULTADOS E DISCUSSÃO

A aplicação do pipeline ao aglomerado aberto NGC 2266 permitiu a obtenção automática dos seus principais parâmetros físicos a partir dos dados fotométricos do SDSS e das isócronas Dartmouth. Os resultados são apresentados a seguir, em ordem direta das etapas implementadas no código.

Estimativa do módulo de distância

A partir do procedimento de pareamento ponto a ponto entre a isócrona teórica e as estrelas observadas, o pipeline calculou o módulo de distância como a mediana dos valores $m - M$ para todas as correspondências válidas. O valor obtido foi:

$$\mu = 14,43,$$

o que corresponde a uma distância heliocêntrica de:

$$d = 7,7 \text{ kpc}.$$

Esse valor encontra-se compatível com estimativas presentes na literatura, que situam o aglomerado entre 7 e 8 kpc. A convergência do método confirma que o pareamento cor–magnitude é adequado para esse tipo de *cluster*, especialmente quando se dispõe de uma sequência principal bem definida, como é o caso do NGC 2266.

Identificação automática do ponto de *turn-off*

A spline ajustada à sequência principal mostrou uma mudança de concavidade claramente identificável no CMD. A derivada segunda da curva apresentou um máximo acentuado na região $g-r \approx 0,38$, o que o pipeline identificou como o ponto de *turn-off*. Esse valor é consistente com isócronas intermediárias (1–2 Gyr), confirmando qualitativamente a idade esperada.

A vantagem do método é a eliminação da subjetividade associada ao ajuste visual, tornando o processo reprodutível e adequado para análises de grandes amostras de aglomerados.

Compatibilidade com isócronas teóricas

A isócrona de 1,2 Gyr e metalicidade solar, quando deslocada pelo módulo de distância estimado, apresentou boa coincidência com a sequência principal observada e com o *turn-off*. A região da gigante vermelha (RGB) mostrou maior dispersão, como esperado em dados fotométricos do SDSS, mas ainda compatível com o ramo ascendente da isócrona dentro das incertezas.

A Figura 6 ilustra essa sobreposição, destacando que:

- A sequência principal inferior apresenta excelente concordância;
- O turno evolutivo coincide com a curvatura prevista pela isócrona;
- A região de RGB é escassamente populosa, mas preserva a tendência teórica.

Diagrama HR e consistência

A conversão automática das magnitudes aparentes em magnitudes absolutas permitiu reconstruir o diagrama HR observado (Figura 7). A forma geral do diagrama reproduz a topologia típica de um aglomerado intermediário: sequência principal extensa, *turn-off* definido e ramo gigante pouco populoso. A sobreposição do modelo confirma que:

- as diferenças entre a isócrona e os pontos observados são pequenas ao longo de toda a sequência principal;
- o valor de μ obtido é consistente tanto no CMD quanto no HR;
- o conjunto de parâmetros idade-distância ajustado pelo pipeline é fisicamente plausível.

Validação da abordagem automatizada



Os resultados foram comparados a ajustes manuais realizados previamente e demonstraram consistência dentro de $\sim 0,1$ mag no módulo de distância e $\sim 0,2$ mag no turn-off. Essas diferenças são esperadas e permanecem dentro do limite introduzido pelas incertezas fotométricas do SDSS.

O método proposto apresenta três vantagens principais:

1. Elimina subjetividade e depende apenas dos dados de entrada.
2. O ajuste completo é realizado em poucos segundos.
3. A estrutura do pipeline é adaptável para outros aglomerados e bancos de isócronas.

Assim, a aplicação ao NGC 2266 demonstra que o pipeline é eficiente, fornece resultados realistas e pode ser empregado tanto em contexto didático quanto em análises preliminares de grandes catálogos estelares.

CONSIDERAÇÕES

O pipeline desenvolvido demonstrou ser uma ferramenta eficiente para a determinação automatizada de parâmetros fundamentais de aglomerados estelares a partir de dados fotométricos. A metodologia integra, em um único fluxo de processamento, a leitura dos dados do SDSS, a construção dos diagramas cor–magnitude e HR, a identificação automática do ponto de turn-off e a comparação quantitativa com isócronas teóricas do Dartmouth Stellar Evolution Database.

A aplicação ao aglomerado NGC 2266 confirmou a capacidade do algoritmo em reproduzir resultados compatíveis com valores encontrados na literatura, tanto para o módulo de distância quanto para a idade estimada. Além disso, o método mostrou-se robusto frente às incertezas fotométricas do SDSS e reprodutível, eliminando a subjetividade associada ao ajuste visual tradicionalmente utilizado em análises de CMD.

Embora promissor, o pipeline ainda pode ser expandido. Melhorias futuras incluem a implementação de técnicas de otimização baseadas em mínimos quadrados, métodos bayesianos, e a leitura automatizada de grades completas de isócronas. Outra possibilidade é a inclusão de estimativa de extinção interestelar, atualmente não tratada pelo algoritmo, além da aplicação sistemática do método a uma amostra maior de aglomerados para validação estatística.

Em síntese, o presente estudo demonstrou que a automação do ajuste de isócronas via Python é uma abordagem viável, reprodutível e de grande potencial tanto para fins didáticos quanto para pesquisas preliminares em evolução estelar e estudos de populações estelares.

REFERÊNCIAS

BINNEY, J.; MERRIFIELD, M. **Galactic Astronomy**. Princeton: Princeton University Press, 1998.



DOTTER, A.; CHABOYER, B.; et al. The Dartmouth Stellar Evolution Database: Models and Isochrones for a Wide Range of Ages and Metallicities. **The Astrophysical Journal Supplement Series**, v. 178, p. 89–101, 2008.

YORK, D. G. et al. The Sloan Digital Sky Survey: Technical Summary. **The Astronomical Journal**, v. 120, n. 3, p. 1579–1587, 2000.

SALARIS, M.; CASSISI, S. **Evolution of Stars and Stellar Populations**. Chichester: Wiley, 2005.

MONTEIRO, H.; DIAS, W. S.; SANTIAGO, B. X. Automated Isochrone Fitting for Open Clusters Using Photometric Data. **Astronomy & Astrophysics**, v. 655, A94, 2021.

DIAS, W. S.; LÉPINE, J. R. D.; ANDRADE, B. G.; et al. Fitting isochrones to open cluster photometric data. **Astronomy & Astrophysics**, v. 539, A79, 2012.

Como citar este artigo (ABNT):

BELLINI, G. A. Desenvolvimento de um pipeline em python para determinação automática da idade e distância de aglomerados estelares a partir de dados fotométricos do Sloan Digital Sky Survey. **Informativo do Observatório Didático de Astronomia (IODA)**, v. 6, n.1, p. 29-43, jun 2026.

