Universidade Federal do Rio de Janeiro
Centro de Ciências Matemáticas e da Natureza
Observatório do Valongo

Classificação de Estrelas de Alta Massa com Aprendizado de Máquina Não Supervisionado

Rodrigo Barros Gonçalves

Rio de Janeiro
Setembro de 2022

Classificação de Estrelas de Alta Massa com Aprendizado de Máquina Não Supervisionado

Rodrigo Barros Gonçalves

Trabalho de conclusão de curso submetido ao Curso de Graduação em Astronomia, Observatório do Valongo, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Astrônomo.

Orientador: Wagner Marcolino

Rio de Janeiro
Setembro de 2022

CIP - Catalogação na Publicação

Gonçalves, Rodrigo Barros
G277c Classificação de estrelas de alta massa com aprendizado de máquina não supervisionado / Rodrigo Barros Gonçalves. -- Rio de Janeiro, 2022.
64 f.

Orientador: Wagner Luiz Ferreira Marcolino.
Trabalho de conclusão de curso (graduação) - Universidade Federal do Rio de Janeiro, Observatório do Valongo, Bacharel em Astronomia, 2022.

1. Aprendizado de máquina. 2. Estrelas de alta massa. 3. Python. 4. k-means. 5. Larguras equivalentes. I. Marcolino, Wagner Luiz Ferreira, orient. II. Título.

Elaborado pelo Sistema de Geração Automática da UFRJ com os dados fornecidos pelo(a) autor(a), sob a responsabilidade de Miguel Romeu Amorim Neto - CRB-7/6283.

UNIVERSIDADE FEDERAL DO RIO DE JANEIRO
CCMN - OBSERVATÓRIO DO VALONGO
DEPARTAMENTO DE ASTRONOMIA

PROJETO FINAL

RELATÓRIO DA COMISSÃO JULGADORA

ALUNO: Rodrigo Barros Gonçalves (DRE 112081403)

TÍTULO DO TRABALHO: "CLASSIFICAÇÃO DE ESTRELAS DE ALTA MASSA COM APRENDIZADO DE MÁQUINA NÃO SUPERVISIONADO"

DATA DA DEFESA: 17 de outubro de 2022 às 13:15 h

MEMBROS DA COMISSÃO JULGADORA:

Dr. Wagner Marcolino – Presidente/Orientador - (OV/UFRJ)

Dr. Paulo Afrânio Lopes - (OV/UFRJ)

Eduardo Machado Pereira

MSc. Eduardo Machado Pereira – (ON/MCTIC)

CANDIDATO: Rodrigo Barros Gonçalves

Rodrigo Barros Gonçalves

Rio de Janeiro, 17 de outubro de 2022.

Prof. Paulo Afrânio Augusto Lopes
Coord. de Grad. do Curso de Astronomia

Dedico este trabalho aos meus pais Nestor e Verônica, ao meu cachorro Pingo, a toda a minha família e, sob o risco de parecer demasiadamente vaidoso, a mim mesmo.

Agradecimentos

O caminho que percorri desde o ingresso na universidade até a conclusão deste trabalho foi recheado de aprendizados sobre o cosmos e sobre mim. Mas também foi inegavelmente longo e difícil. Portanto, é natural que eu tenha sido apoiado das mais diversas formas por um sem-número de pessoas, muitas com as quais ainda tenho contato, e outras já não mais, seja porque nos distanciamos, ou porque infelizmente partiram. De todo modo, quero usar este espaço para demonstrar um pouco da minha gratidão a ao menos algumas das pessoas que me apoiaram e incentivaram ao longo desta trajetória, pois sem elas imagino que nada disso teria sido possível.

Quero dar destaque a gratidão que tenho à minha mãe Verônica e ao meu pai Nestor. Deles recebi incentivo e apoio constante durante os anos não só da graduação, mas também da vida. Foi graças a eles que pude seguir no meu interesse pela astronomia há tantos anos, e assim, considero-me particularmente afortunado por ter a oportunidade de deixar seus nomes nas linhas deste texto, ao mesmo tempo que fico frustrado por sentir que seja impossível colocar apropriadamente em palavras uma fração sequer do amor que sinto por eles.

A quantidade de pessoas que gostaria de colocar aqui ao pensar em minha família é muito grande e, portanto, deixo este parágrafo para mencionar apenas algumas. Meu antigo cachorro Pingo não é uma pessoa, mas cresci com ele, e devido ao apego que ainda carrego comigo, quero destacá-lo aqui juntamente com minha avó Marlene, com quem tanto conversei sobre os astros, com meu avô Anael que também sempre me perguntava sobre a graduação, meus primos Filipe, Thaiana, Bianca, Carolina, Bárbara e Beatriz, minha madrinha Mônica, meu padrinho Silmar e meus tios e tias Carla, José Luiz, Renata, Alexandre, Ricardo e Simone. Além disso, meu tio em segundo grau Ronaldo e minha prima em segundo grau Flavia, que também me incentivaram cada um à sua maneira.

Neste último parágrafo quero agradecer a todos que conheci durante os anos em que estudei no Colégio de Aplicação do Instituto Isabel, especialmente ao meu amigo Vitor. Dentre os que conheci na UFRJ, quero agradecer aos colegas do Tapirapé, em especial meus orientadores Wagner e Helio, professor Gustavo, e colegas Matheus, Júlia, Ellen e Eduardo. Outros que conheci na universidade e que quero mencionar aqui são os colegas Pedro Nogueira, Ana Posses, Aline, Karícia, Douglas Rodrigues, Bruno e Natália. Finalmente, quero também mencionar aqui meu psicólogo Marco Antonio que também foi determinante na minha caminhada.

"Nós estamos, em nossa relação com a vida, como um peixinho num imenso oceano, em maravilhosa fruição. Nunca vai ocorrer a um peixinho que o oceano tem que ser útil, o oceano é a vida."

Ailton Krenak,
A vida não é útil

Resumo

Classificação de Estrelas de Alta Massa com Aprendizado de Máquina Não Supervisionado

Rodrigo Barros Gonçalves

Orientador: Wagner Marcolino

A classificação espectral de estrelas de alta massa fornece diretamente uma ideia do status evolutivo de determinado objeto e de parâmetros físicos importantes, como a temperatura e luminosidade, e até mesmo da velocidade de rotação superficial. Neste trabalho, foram utilizadas técnicas de aprendizado de máquina não supervisionado para atacar o problema de classificação espectral em estrelas de alta massa com duas motivações principais: (i) aprender astrofísica de estrelas de alta massa e (ii) aprender técnicas de aprendizado de máquina não supervisionado. A implementação de algoritmos de aprendizado de máquina pode se tornar uma ferramenta muito útil para a extração de informações relevantes de grandes amostras e surveys (e.g., Gaia EDR3), não somente para fins de classificação espectroscópica, mas também com fotometria e outras grandezas físicas medidas (e.g., velocidades radiais, composições químicas). Uma amostra de 606 espectros de estrelas de tipo espectral O oriundas do mais amplo catálogo disponível - "*Galactic O-Star Catalog*" -, e também de diversas fontes listadas em Martins (2018) foi reunida. Larguras equivalentes de linhas espectrais em cada objeto de nossa amostra foram medidas. Em seguida, o espaço de largura de linhas (1 ponto no espaço N-dimensional correspondendo a 1 estrela com N linhas medidas) e suas aglomerações foram analizadas por meio de um algoritmo de aprendizado de máquina não supervisionado (*k-means clustering*). Posteriormente, a fim de testarmos efeitos relacionados ao tamanho da amostra, elaboramos uma estratégia para gerar estrelas artificiais, i.e., novas larguras equivalentes, a partir das observadas. Ao todo, ficamos com 47.000 estrelas e a análise anterior foi repetida. Nos testes conduzidos com a amostra original de 606 estrelas, encontramos que a qualidade da classificação deixa a desejar. Mesmo quando utilizamos apenas 3 *features* (com o silhouette *score* sugerindo uma boa classificação), a diferenciação das classes nos *clusters* encontrados permaneceu insatisfatória. Nos testes realizados com a amostra maior, os resultados melhoraram substancialmente, com exceção do caso em que usamos as 3 *features*, onde houve piora. Isso provavelmente se deve ao critério de junção das classes de luminosidade que adotamos, o que implica numa revisão e aprimoramento da análise destas classes. Em suma, verificamos que algoritmos não supervisionados podem fornecer uma classificação espectral prévia satisfatória em novos conjuntos de dados desde que a amostra seja suficientemente grande, o que já é bastante útil para análises posteriores.

palavras-chave — Estrelas de Alta Massa, Aprendizado de Máquina, k-means, Python

Abstract

Classification of Massive Stars With Unsupervised Machine Learning

Rodrigo Barros Gonçalves

Advisor: Wagner Marcolino

The spectral classification of massive stars directly provides an idea of the evolutionary status of a given object and of important physical parameters, such as temperature and luminosity, and even of the velocity of superficial rotation. Is this work, unsupervised machine learning techniques were used to approach the problem of spectral classification of massive stars with two main motivations: (i) learn astrophysics of massive stars and (ii) learn unsupervised machine learning techniques. The implementation of machine learning algorithms might become a very useful tool for the extraction of relevant information from large samples and surveys (e.g., Gaia EDR3), not only for purposes of spectroscopic classification, but also with photometry and other measured physical quantities (e.g., radial velocities, chemical compositions). A sample of 606 spectra of O-type stars taken from the largest available catalog - "Galactic O-Star Catalog" -, as well as from various other sources listed in Martins (2018) was gathered. Equivalent widths of many spectral lines in each object of the sample were measured. Subsequently, the space of line widths (e.g., 1 point in the N-dimensional space corresponding to 1 star with N measured lines) and their clustering were analyzed by way of an unsupervised machine learning algorithm (k-means clustering). Subsequently, in order to test the effects of the sample size, we elaborated a strategy to generate artificial stars, i.e., new equivalent widths, from the observed ones. We arrived at a total of 47,000 stars and the previous analysis was repeated. In the tests conducted with the original sample of 606 stars, we found that the quality of the classification is lacking. Even when we utilized only 3 features (with the silhouette score hinting towards a good classification), the differentiation between the classes remained unsatisfactory. In the tests carried out with the larger sample, the results improved substantially, with the exception of the case in which we use 3 features, where they worsened. This is likely due to the adopted criterion of joining luminosity classes, which suggests reviewing and improving the analysis of such classes. In short, we found that unsupervised algorithms can provide a satisfactory prior spectral classification in new data sets as long as the sample is sufficiently large, which is already quite useful for further analysis.

keywords — Massive Stars, Machine Learning, k-means, Python

Lista de Figuras

1.1 Espectros de estrelas com diferentes tipos espectrais 17

1.2 Comparativo de linhas de absorção em estrelas com classes de luminosidade distintas 18

1.3 Diagrama da correspondência entre uma linha de absorção qualquer e um retângulo de mesma área com sua respectiva largura equivalente 19

1.4 Comparação do comportamento das linhas He I λ4471 e He II λ4542 em estrelas O 20

1.5 Logaritmo da razão entre as larguras equivalentes de He I λ4471 e He II λ4542 em função do tipo espectral 21

1.6 Comportamento da linha He II λ4686 em diferentes classes de luminosidade . . . 22

1.7 Comportamento das linhas He II λ4686, He I λ4713, Si IV λ4089 e He I λ4026 em estrelas O9-O9.7 23

2.1 Etapas da identificação bem-sucedida de *clusters* em dados ilustrativos 32

2.2 Identificação insatisfatória de *clusters* em dados ilustrativos 33

3.1 Silhouette score médio em função de k da amostra de 606 estrelas 37

3.2 Silhouette score médio em função de k da amostra de 47.000 estrelas 37

3.3 Silhouette score médio em função de k da amostra de 606 estrelas para os casos de menor dimensionalidade 38

3.4 Silhouette score médio em função de k da amostra de 47.000 estrelas para os casos de menor dimensionalidade 39

3.5 Histogramas dos clusters resultantes. 606 estrelas, 7 *features* ($k = 47$). 41

3.6 Histogramas dos clusters resultantes. 47.000 estrelas, 7 *features* ($k = 47$). 45

3.7 Histogramas dos clusters resultantes. 606 estrelas, 4 *features* ($k = 16$). 50

3.8 Histogramas dos clusters resultantes. 606 estrelas, 3 *features* ($k = 3$). 51

3.9 Histogramas dos clusters resultantes. 47.000 estrelas, 4 *features* ($k = 16$). 52

3.10 Histogramas dos clusters resultantes. 47.000 estrelas, 3 *features* ($k = 3$). 53

3.11 Gráficos dos silhouette *scores* individuais de cada *cluster* para os casos de 606 e 47.000 estrelas 55

Lista de Tabelas

2.1 Quantidade de Estrelas Observadas por Tipo Espectral 29

A.1 Amostragem dos Dados Observados 62

A.2 Amostragem dos Dados Artificiais 62

Lista de Abreviaturas e Siglas

- Gaia EDR3: Gaia Early Data Release 3
- GOSC: Galactic O-Star Catalog
- MK: Morgan-Keenan. Refere-se ao sistema de classificação espectral de estrelas
- SDSS-V: Sloan Digital Sky Survey V
- W_λ: Largura Equivalente, do inglês *equivalent width*
- FITS: Flexible Image Transport System. Formato de arquivo digital amplamente utilizado em astronomia
- Estrelas OC e ON: Estrelas O ainda pouco compreendidas cujas características marcantes são a morfologia das linhas de carbono e nitrogênio, sua evolução química e similaridades com estrelas Wolf-Rayet

Sumário

1 Introdução **15**

1.1 O Sistema de Classificação MK 15

1.2 Classificação Espectral de Estrelas O 19

1.2.1 Subtipos Espectrais 19

1.2.2 Classes de Luminosidade 22

1.3 Aprendizado de Máquina Não Supervisionado 24

2 Dados e Metodologia **26**

2.1 Dados Observacionais 26

2.1.1 Galactic O-Star Catalog 27

2.1.2 Amostra de Martins (2018) 27

2.2 Dados Artificiais 30

2.3 Classificação 31

2.3.1 k-means 31

3 Resultados **35**

3.1 Aplicação do Método k-means 35

3.1.1 Escolha de Features 35

3.1.2 Definição do Hiperparâmetro k 36

3.1.3 Clusters Obtidos 40

3.2 Discussão de Problemas 56

4 Conclusão e Perspectivas **57**

Bibliografia **58**

Apêndice A Tabela dos dados utilizados **62**

Apêndice B O algoritmo k-means **63**

Capítulo 1

Introdução

Neste trabalho de conclusão de curso, investigamos o espectro de estrelas de alta massa através de técnicas de aprendizado de máquina não supervisionado. Em particular, focamos na classificação espectral desses objetos, assunto que possui questões em aberto e trabalhos recentes na literatura (e.g., Martins 2018, Maravelias et al. 2022). Portanto, inicialmente, iremos revisar conceitos fundamentais de classificação espectral, dando ênfase às estrelas de alta massa do tipo O. Em seguida, discutiremos brevemente o que é aprendizado de máquina supervisionado e não supervisionado (*machine learning*). Por último, descreveremos a estrutura do restante do projeto.

1.1 O Sistema de Classificação MK

As atividades de observação e estudo da radiação eletromagnética emitida por objetos astronômicos possibilitam a descoberta de suas características (e.g., massa, metalicidade, distância), e, por meio destas, sua eventual diferenciação e classificação. Neste sentido, a espectroscopia é uma das técnicas observacionais mais proveitosas, pois a análise de espectros fornece uma riqueza de dados que permite, dentre muitas coisas, a classificação de estrelas com o sistema Morgan-Keenan (MK).

O sistema MK atribui classes de luminosidade I, II, III, IV e V (supergigantes, gigantes brilhantes, gigantes, subgigantes e anãs, respectivamente) ao sistema de Harvard, que, de acordo com a temperatura superficial das estrelas, as classifica das mais quentes para as mais frias, tradicionalmente, nos tipos espectrais O, B, A, F, G, K e M, cada um com subdivisões que vão de 0 a 9 em ordem decrescente de temperatura. É importante ressaltar que existem tipos espectrais adicionais para objetos menos comuns, como estrelas Wolf-Rayet, bem como classes de luminosidade adicionais para hipergigantes, subanãs e anãs brancas, e subdivisões na classe das supergigantes para diferentes luminosidades. Portanto, a análise de espectros é indispensável quando o objetivo é classificar estrelas com um sistema associado a parâmetros obtidos a partir de linhas espectrais.

A figura 1.1 evidencia a diferença entre espectros de estrelas anãs de tipos distintos. Estrelas quentes apresentam linhas de hélio neutro ou ionizado, bem como linhas de hidrogênio fracas, devido à ionização. Em estrelas de temperaturas mais intermediárias, as linhas de Balmer tornam-se mais fortes, e linhas de metais ionizados (como Ca II) começam a surgir, devido à presença mais significativa de certos íons na atmosfera estelar. Nas mais frias as linhas de Balmer somem, pois não há átomos de hidrogênio excitados o suficiente para absorver os fótons nestas frequências. Entretanto, as bandas de absorção de óxido de titânio apresentam-se com muita intensidade. Na figura estão representados apenas espectros de estrelas anãs, mas o mesmo critério é válido para diferenciar e classificar estrelas de outras classes de luminosidade.

Mesmo que sejam do mesmo tipo espectral, estrelas de classes de luminosidade distintas também apresentam diferenças em seus espectros. O raio elevado de uma estrela, por exemplo, reflete-se em sua luminosidade, que também pode ser considerada elevada se comparada à luminosidade de estrelas menores de mesma temperatura. Essa variação na luminosidade entre estrelas pode ser percebida no espectro medindo-se o alargamento das linhas. O alargamento ou estreitamento é consequência da densidade nas atmosferas estrelares. Em estrelas de raio reduzido (luminosidade reduzida), a fotosfera está próxima ao núcleo e, portanto, a gravidade superficial é mais intensa do que em estrelas de maior raio. Se a gravidade superficial é mais intensa, a densidade é maior e o campo elétrico médio devido às partículas é maior. Isto intensifica o efeito Stark, que aumenta o número de transições das partículas e, consequentemente, a

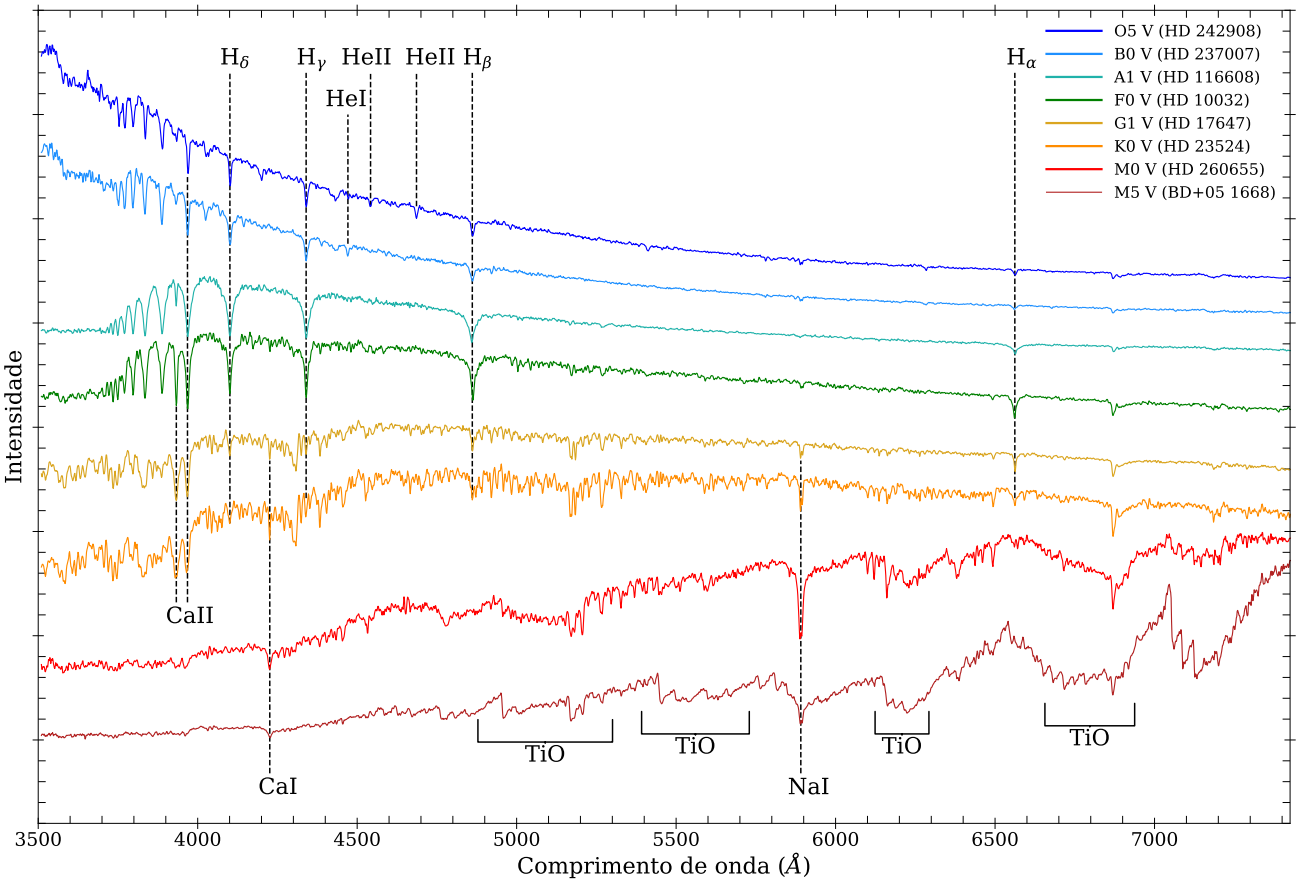


Figura 1.1: Espectros de estrelas com diferentes tipos espectrais – Gráfico de intensidade da radiação por comprimento de onda que destaca algumas linhas de absorção em espectros de anãs de diferentes tipos. Os tipos espectrais estão indicados na legenda e associados a cores distintas. As linhas tracejadas indicam linhas de absorção, e colchetes pretos indicam algumas das bandas de óxido de titânio. O gráfico é baseado na imagem 7-9 de Seeds & Backman (2011) e a seleção das estrelas foi feita com base na biblioteca espectrofotométrica de Jacoby et al. (1984), cujos FITS estão disponíveis no arquivo[1] do STScI.

absorção de fótons em comprimentos de onda próximos dos comprimentos de onda originais das linhas espectrais, causando seu alargamento. Em estrelas com raio comparativamente maior, as linhas não sofrem o alargamento na mesma intensidade.

A Figura 1.2 ilustra este fenômeno. Embora de modo sutil, é possível notar que as linhas de absorção da estrela anã são mais largas do que as linhas das estrelas gigante e supergigante. Vale ressaltar que outros fatores também podem causar alterações na largura de linhas espectrais, como alargamento doppler e alta rotação.

[1] https://www.stsci.edu/hst/instrumentation/reference-data-for-calibration-and-tools/astronomical-catalogs/jacoby-hunter-christian-atlas

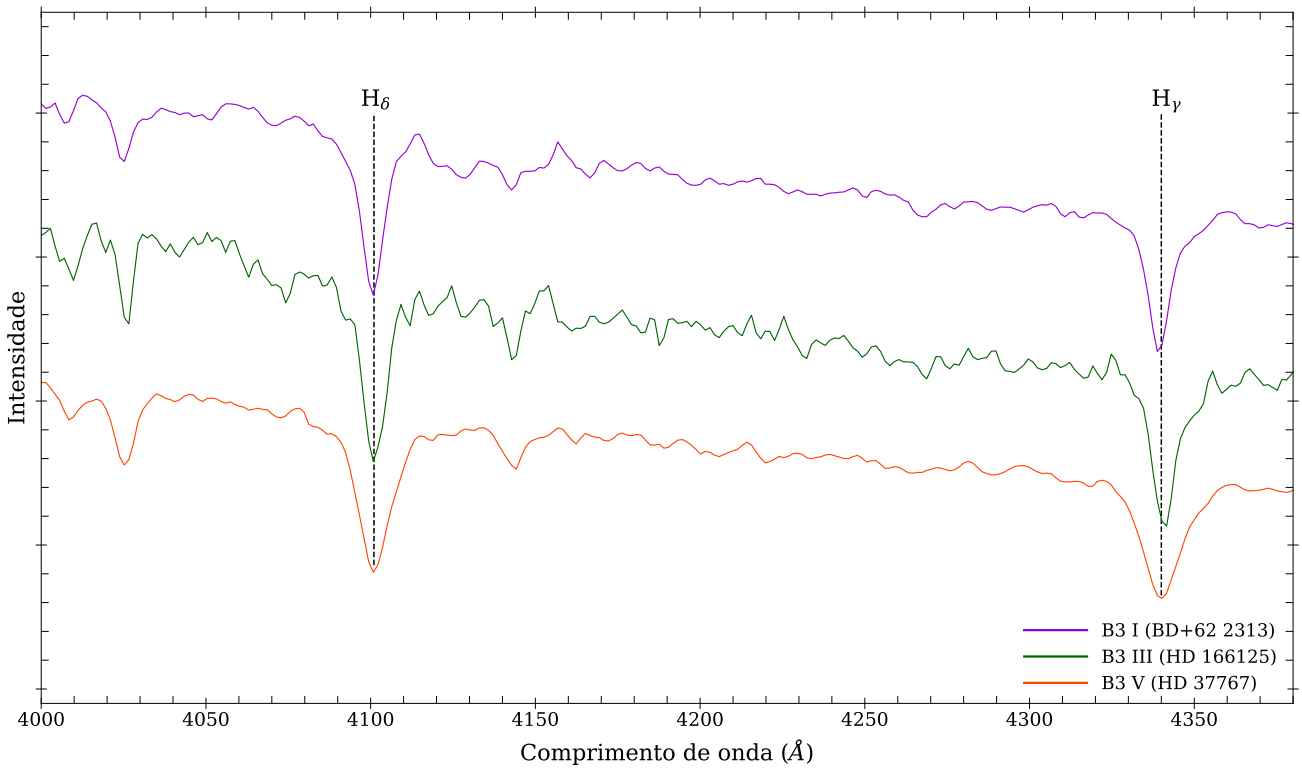


Figura 1.2: Comparativo de linhas de absorção em estrelas com classes de luminosidade distintas – Gráfico da intensidade de radiação por comprimento de onda que mostra o mesmo intervalo nos espectros de uma supergigante, gigante, e anã de mesmo subtipo espectral. As classes estão indicadas na legenda e as linhas tracejadas indicam as linhas que sofrem alargamento. A seleção das estrelas foi feita de modo igual à Figura 1.1.

Para os fins de classificação deste trabalho, será necessário obter medidas quantitativas que reflitam a intensidade das linhas presentes nos espectros. Isso é feito através da medição das larguras equivalentes das linhas espectrais. A largura equivalente é definida como a largura de um retângulo cuja área é igual à área delimitada por um perfil de linha e seu contínuo.

No contexto da classificação espectral, essa área é a região acima (abaixo) da curva descrita pela linha de absorção (emissão), cuja largura equivalente tem unidade de comprimento de onda, como mostra a Figura 1.3. Esse conceito é expresso matematicamente através da definição 1.1, onde W_λ é a largura equivalente, F_c é o fluxo do contínuo, F_λ é o fluxo nos pontos da curva descrita pela linha espectral, e λ_1 e λ_2 são os limites de integração, que definem o intervalo que compreende totalmente a linha em questão. Como todos os espectros utilizados nesse trabalho estão normalizados, calculamos as larguras com base na Equação 1.2.

$$W_\lambda \equiv \int_{\lambda_1}^{\lambda_2} \frac{F_c - F_\lambda}{F_c} d\lambda \tag{1.1}$$

Em um espectro normalizado,

$$F_c = 1$$

então

$$W_\lambda = \int_{\lambda_1}^{\lambda_2} (1 - F_\lambda) d\lambda \tag{1.2}$$

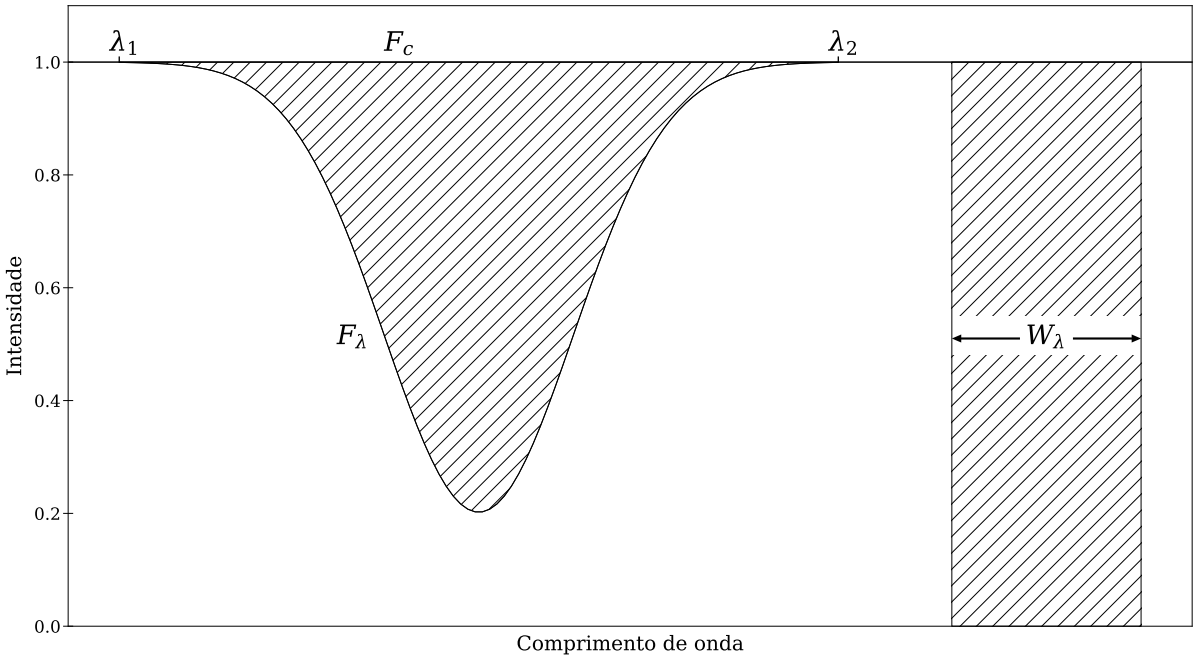


Figura 1.3: Diagrama da correspondência entre uma linha de absorção qualquer e um retângulo de mesma área com sua respectiva largura equivalente – As regiões hachuradas representam as áreas em questão. λ_1, λ_2, F_c, F_λ e W_λ são respectivamente os limites mínimo e máximo da linha de absorção, o fluxo do contínuo, o fluxo da linha e a largura do retângulo (largura equivalente).

1.2 Classificação Espectral de Estrelas O

1.2.1 Subtipos Espectrais

O critério largamente adotado para determinar os subtipos de estrelas O envolve o cálculo da razão entre as larguras equivalentes de He I λ4471 e He II λ4542 (Conti & Alschuler, 1971; Morgan & Keenan, 1973; Mathys, 1988). Conhecendo-se o comportamento da intensidade de ambas as linhas conforme a temperatura efetiva varia, é possível associá-la aos valores resultantes da razão de larguras equivalentes, e assim determinar os subtipos.

Como o grau de ionização de átomos está relacionado à temperatura, estrelas mais tardias apresentam He II λ4542 fraco ou ausente e He I λ4471 mais intenso. Com o aumento da temperatura, a intensidade de He I λ4471 decresce rapidamente e a de He II λ4542 aumenta até ficar aproximadamente constante em torno de 40.000 K. Esta relação entre temperatura e largura equivalente das linhas de hélio pode ser verificada nas figuras 15 e 17 de Auer & Mihalas (1972). A Figura 1.4 mostra esta mesma relação por meio da comparação visual direta entre os espectros de anãs O.

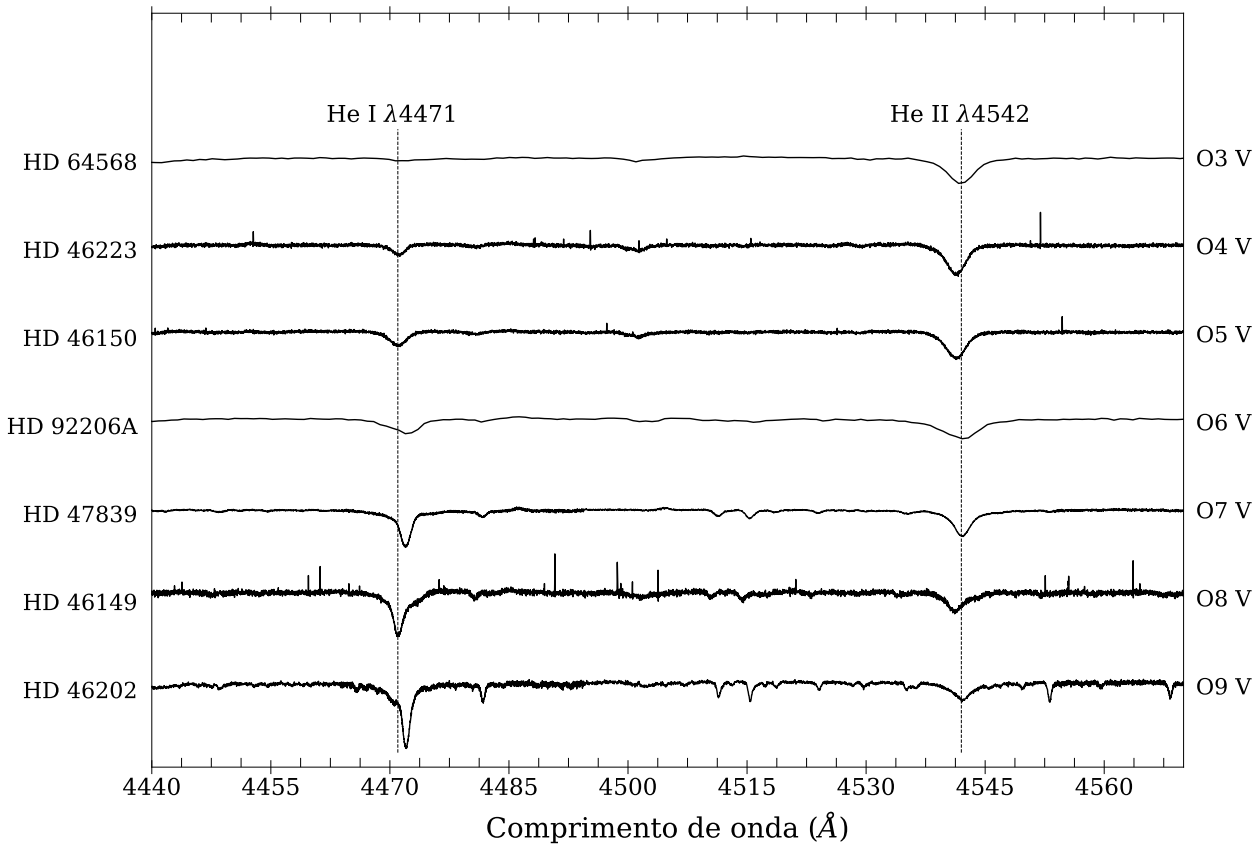


Figura 1.4: Comparação do comportamento das linhas He I λ4471 e He II λ4542 em anãs O – Gráfico de espectros de diversas anãs O no intervalo que compreende as linhas de absorção He I λ4471 e He II λ4542, indicadas por linhas tracejadas de modo aproximado. Os nomes e classificações das estrelas estão dispostos nas extremidades esquerda e direita, respectivamente. Os espectros utilizados são da nossa amostra (mais detalhes no Capítulo 2).

Então, para valores maiores que 40.000 K, que correspondem aproximadamente a estrelas mais quentes que O7, apenas He I λ4471 apresenta forte dependência de temperatura, de modo que o valor da razão entre as larguras equivalentes permite determinar com mais confiança os subtipos nesta faixa (Conti & Frost, 1977).

Na Figura 1.5 é possível verificar a relação entre a razão das larguras e os tipos espectrais. Para refinar a classificação de estrelas mais frias (O8, O8.5, O9, O9.2, O9.5 e O9.7), três novas razões de larguras equivalentes (W_λ) definidas por Sota et al. (2011) são utilizadas: W_λ(He I λ4144)/W_λ(He II λ4200), W_λ(He I λ4388)/W_λ(He II λ4542) e W_λ(Si III λ4552)/W_λ(He II λ4542). Para definir os subtipos O8, O8.5, O9.5 e O9.7, as três razões podem ser utilizadas seguindo critérios particulares. Mas apenas a razão W_λ(He I λ4144)/W_λ(He II λ4200) permite diferenciar as subclasses O9 e O9.2 (Martins, 2018).

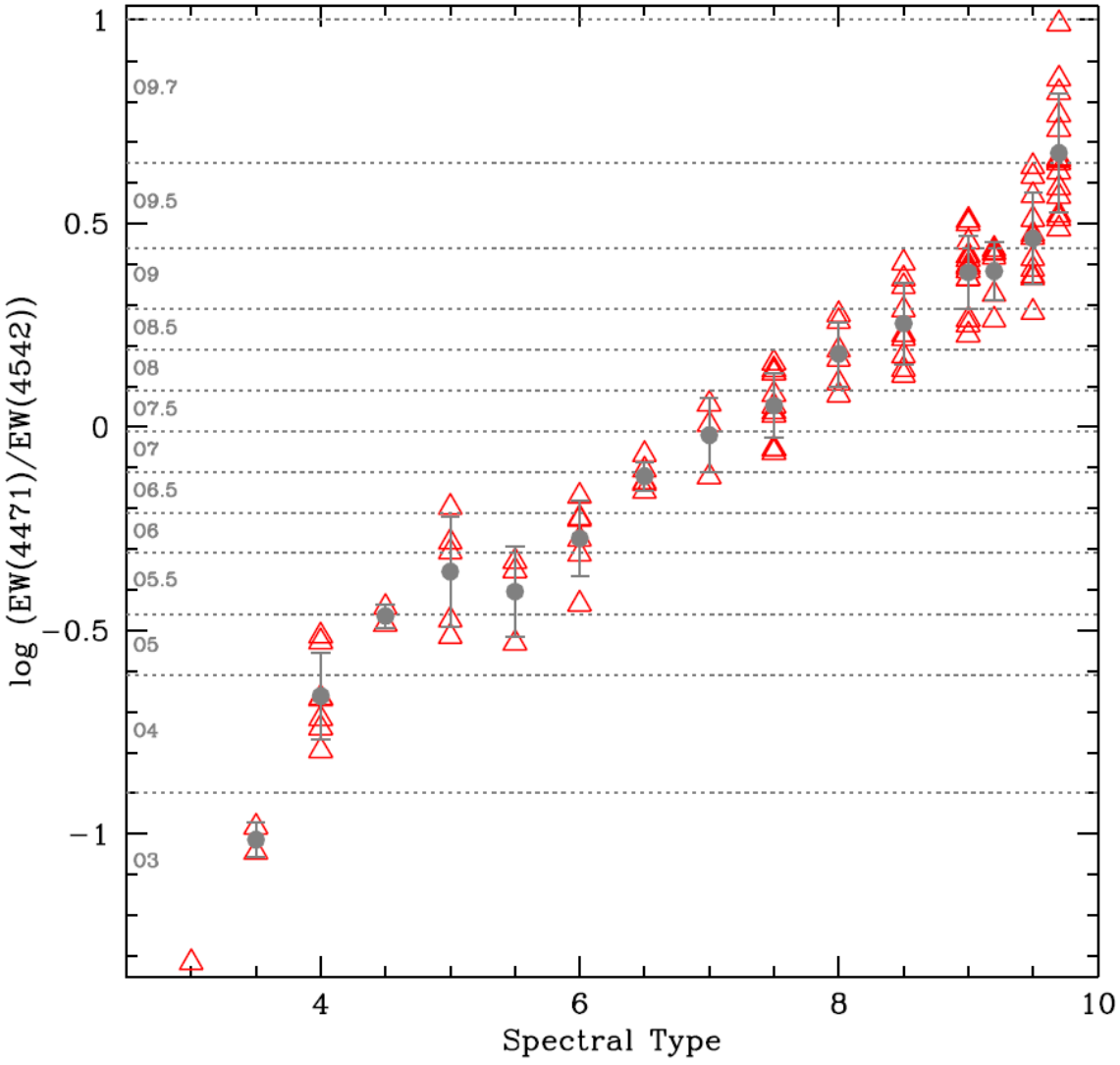


Figura 1.5: Logaritmo da razão entre as larguras equivalentes de He I λ4471 e He II λ4542 em função do tipo espectral – Elementos de uma amostra de 105 estrelas são representados pelos triângulos vermelhos. Pontos cinzas e barras de erro indicam a média e dispersão. As linhas pontilhadas indicam os valores-limite para os tipos espectrais O3 a O9.7. Figura extraída de Martins (2018).

1.2.2 Classes de Luminosidade

Para caracterizar as classes de luminosidade, o critério adotado por Martins (2018) é seguido, no qual métodos distintos são usados para classificar as estrelas do intervalo O3-O8.5 em classes de luminosidade I, II-III-IV (agrupadas) e V, e estrelas do intervalo O9-O9.7 em todas as classes de luminosidade. Para estrelas do intervalo O3-O8.5, a largura equivalente da linha He II λ4686 é usada como critério de classificação. Entretanto, pode não ser fácil diferenciar as classes usando o mesmo método para todos os subtipos espectrais. Devido a isso, definem-se diferentes intervalos de valores da largura de He II λ4686 para classificar estrelas O3-O7.5 e estrelas O8-O8.5. Os detalhes desta diferenciação podem ser encontrados nas subseções 3.2.1 e 3.2.2 de Martins (2018). Optamos por omiti-los uma vez que o método de aprendizado de máquina aqui utilizado independe desta informação. Na Figura 1.6, estrelas foram selecionadas para demonstrar a relação da intensidade da linha He II λ4686 com as classes de luminosidade. Estrelas de classes menos luminosas apresentam a linha em absorção, que gradativamente entra em emissão conforme a classe torna-se mais luminosa. Em termos de largura equivalente, quanto maior a absorção (emissão), mais positivo (negativo) é o valor encontrado.

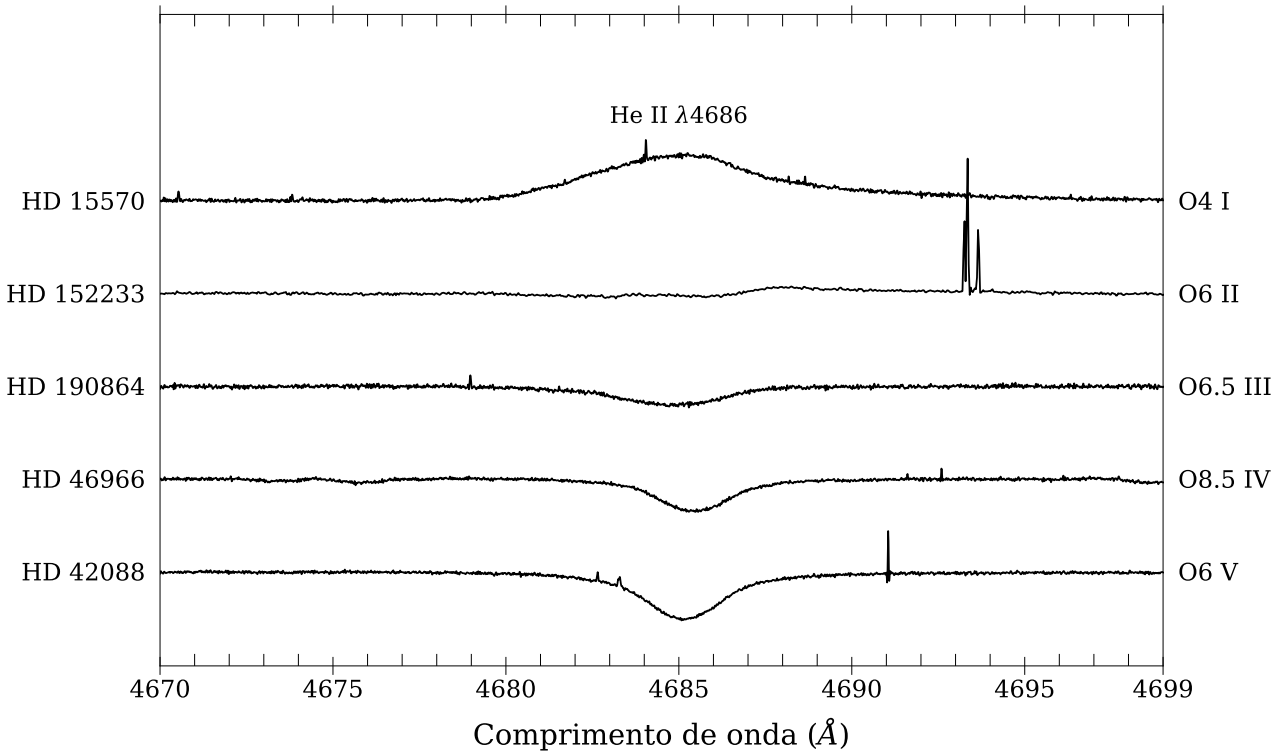


Figura 1.6: Comportamento da linha He II λ4686 em diferentes classes de luminosidade – Espectros de diversas estrelas O no intervalo que compreende a linha de absorção He II λ4686. Os nomes e classificações das estrelas estão dispostos nas extremidades esquerda e direita, respectivamente. Os espectros utilizados são da nossa amostra (mais detalhes no Capítulo 2).

A razão pela qual este critério agrupa as estrelas das classes II-III-IV é a ambiguidade dos valores de suas respectivas larguras equivalentes. Estrelas com a largura equivalente da linha de He II λ4686 nos intervalos $[0.4, 0.6]$ e $[-0.2, 0]$ não têm identificação unívoca, uma vez que estes intervalos definem grande parte das estrelas das classes II, III e IV, especialmente o primeiro (detalhes na Figura 3 de Martins, 2018). Por isso, apenas uma comparação direta com um espectro padrão é capaz de remover a ambiguidade e fornecer a classificação exata.

No intervalo O9-O9.7, duas razões distintas são usadas para definir as classes: W_λ(He II λ4686)/W_λ(He I λ4713) e W_λ(Si IV λ4089)/W_λ(He I λ4026). A primeira em geral apresenta valores entre 0 e 1 para supergigantes, crescendo de maneira aproximadamente monotônica nas classes menos luminosas. Já a segunda tem valor em torno de 1 em supergigantes, e decresce monotonicamente até atingir valores entre 0.2 e 0.5 nas anãs. Esse comportamento pode ser examinado com mais detalhes na subseção 3.2.3 de Martins (2018). Através da Figura 1.7, é possível notar como a variação de intensidade das linhas é refletida nos valores das razões. A linha He II λ4686 é mais fraca que He I λ4713 nas supergigantes, mas gradativamente torna-se mais intensa. Além disso, embora as linhas Si IV λ4089 e He I λ4026 tenham intensidades parecidas nas supergigantes, a linha de hélio torna-se mais intensa nas classes seguintes.

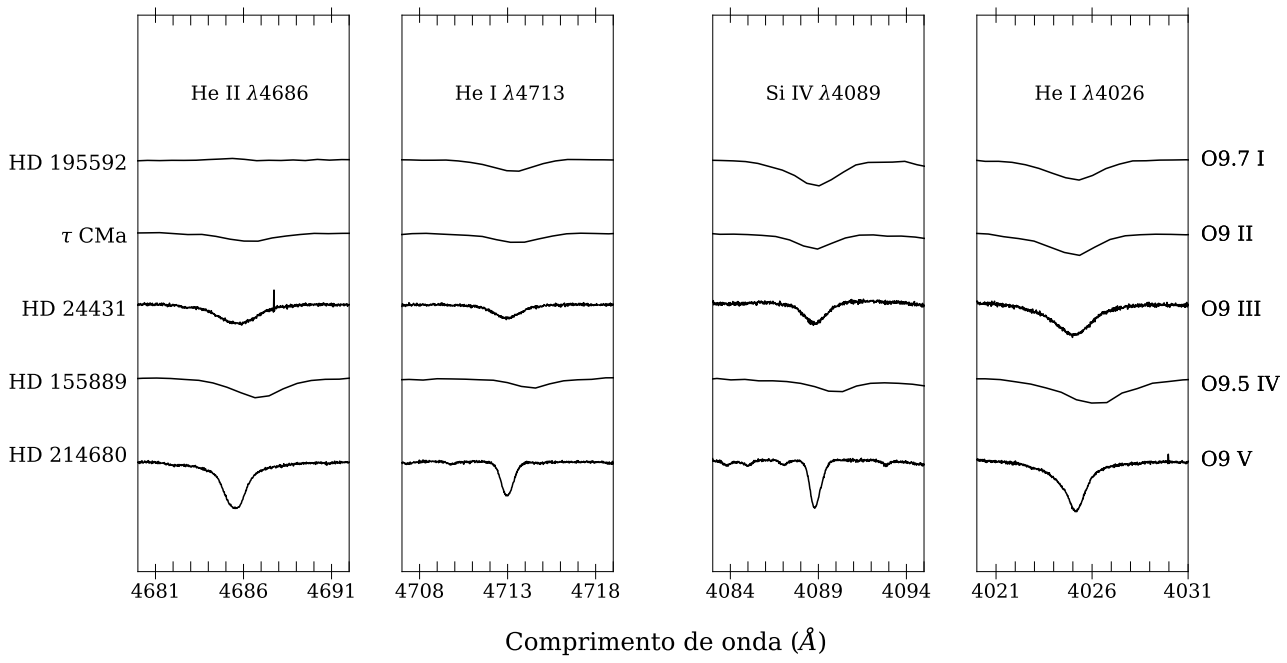


Figura 1.7: Comportamento das linhas He II λ4686, He I λ4713, Si IV λ4089 e He I λ4026 em estrelas O9-O9.7 – Intervalos de espectros contendo linhas usadas na atribuição das classes de luminosidade. Os nomes e classificações das estrelas estão dispostos nas extremidades esquerda e direita, respectivamente. Os espectros utilizados são da nossa amostra (mais detalhes no Capítulo 2).

Conhecendo-se as linhas necessárias para realizar a classificação de estrelas O, já é possível utilizar algoritmos de aprendizado de máquina para executar tal tarefa. Como dito anteriormente, desde que se saiba que é possível fazer a diferenciação das classes com as linhas escolhidas, não há a necessidade de que o algoritmo conheça explicitamente os valores e intervalos das larguras equivalentes que correspondem a cada tipo espectral ou classe de luminosidade, devido à natureza da classificação não supervisionada. Na seção seguinte veremos mais detalhes a respeito deste tipo de algoritmo de modo a esclarecer esse e outros pontos de interesse.

1.3 Aprendizado de Máquina Não Supervisionado

Algoritmos de aprendizado de máquina são capazes de tomar decisões de natureza classificatória e preditiva com base num conjunto de dados (*dataset*) que lhes é apresentado, de modo que, em geral, quanto mais volumoso o *dataset*, mais confiáveis ou desejáveis são os resultados obtidos. O aprendizado de máquina supervisionado é uma abordagem bastante popular que envolve a divisão do *dataset* em dois grupos distintos. Um dos grupos tem a finalidade de treinar o algoritmo a identificar os padrões e características dos dados em questão. O segundo grupo contém o resto dos dados, de características ainda desconhecidas. Cabe então ao algoritmo, com base no treinamento, realizar a tarefa de classificação dos novos dados.

Numa tarefa de classificação de estrelas O, por exemplo, seria necessário treinar o algoritmo apresentando a ele estrelas de cada tipo espectral e classe de luminosidade com as respectivas larguras equivalentes e razões de interesse (mencionadas ao longo deste capítulo), de modo que ao se deparar com uma estrela cuja classificação é desconhecida, ele possa atribuir um subtipo espectral e uma classe de luminosidade a ela com base apenas nos valores com os quais foi treinado. No jargão da área de pesquisa de aprendizado de máquina, as classificações que podem ser atribuídas aos dados são chamadas *labels*, e os valores associados à cada elemento do *dataset* são conhecidos como *features*. Portanto, cada uma das estrelas possui *features* (larguras equivalentes e razões de larguras) que definem a qual *label* (classe) pertencem.

Os algoritmos não supervisionados dispensam a etapa de treino, e isto marca a principal diferença destes para os supervisionados. Consequentemente, numa tarefa de classificação de estrelas O, o algoritmo analisa o *dataset* integralmente, sem dividi-lo em grupos de treino e teste, e então identifica os grupos de estrelas com base na descoberta de padrões e semelhanças entre os valores das larguras equivalentes. Caso o número de *features* seja grande a ponto de aumentar muito o custo computacional da tarefa, pode-se optar pela utilização de algoritmos de redução de dimensionalidade para identificar apenas as *features* mais relevantes e descartar o resto. O resultado da classificação incluirá, portanto, diferentes grupos cujas estrelas o algoritmo julga semelhantes entre si, mas sem a informação sobre a classe de luminosidade e subtipo espectral a qual pertencem, afinal, o algoritmo nunca foi treinado para realizar tal tarefa.

Embora a classificação realizada por este método apresente resultados mais limitados em comparação com métodos supervisionados, em muitos casos sua utilização pode ser preferível. Para treinar um algoritmo, por exemplo, é necessário que os dados tenham sido previamente classificados, e nem sempre esse tipo de informação estará disponível, principalmente considerando o atual contexto de surveys que disponibilizam volumes enormes de dados (e.g., Gaia, SDSS). Além disso, como algoritmos não-supervisionados não recebem treinamento, não ficam restritos apenas a estratégia de classificação para a qual foram treinados. Logo, há a possibilidade de que padrões e relações inesperadas sejam encontradas. Naturalmente, tais relações podem não ser relevantes para o objetivo e devem ser analisadas posteriormente.

Até aqui, apresentamos os conceitos fundamentais relacionados à classificação espectral e ao aprendizado de máquina. O restante deste projeto final está organizado da seguinte forma: no Capítulo 2 apresentamos os dados observacionais reunidos para a análise e a metodologia utilizada. Discutimos a amostra total, os telescópios, espectrógrafos e resoluções utilizadas em cada conjunto de dados, bem como apresentamos o critério utilizado e a motivação para gerar dados artificiais. Introduzimos em seguida o método escolhido para análise – k-means – baseado em "*clustering*" dos parâmetros escolhidos. No Capítulo 3, discutiremos o resultado da aplicação do algoritmo em dados reais e artificiais, ou seja, nas larguras equivalentes medidas em todas as estrelas de nossa amostra, bem como em larguras equivalentes geradas com base nas larguras reais. No Capítulo 4, discutiremos as principais conclusões do trabalho e futuras perspectivas de aprimoramento dos resultados e solução dos problemas encontrados.

Capítulo 2

Dados e Metodologia

2.1 Dados Observacionais

No que diz respeito aos dados reais utilizados neste trabalho, utilizamos 606 espectros de estrelas O. Deste total, 507 foram obtidos a partir do download de arquivos FITS disponíveis na página do Galactic O-Star Catalog[1] (Maíz Apellániz et al., 2013), e os 99 restantes foram obtidos a partir de diversas fontes listadas na tabela A.1 de Martins (2018). Neste trabalho, as subclasses de luminosidade Ia, Ib e Iab são tratadas apenas como I; também não é utilizada a diferenciação com base no critério "f", que caracteriza a intensidade de emissão e absorção de nitrogênio e hélio. A partir dos dados obtidos, as larguras equivalentes foram extraídas automaticamente através de um código escrito em Python. Os comprimentos de onda máximo e mínimo que delimitam as linhas espectrais de interesse foram definidos com base nas estrelas de maior rotação, identificadas pelo qualificador "(n)". Em seguida, as linhas foram integradas e as larguras equivalentes obtidas, conforme explicado no fim da Seção 1.1. Nas duas subseções seguintes serão apresentados mais detalhes a respeito da seleção das estrelas e de suas respectivas fontes.

[1] https://gosc.cab.inta-csic.es/galactic-o-star-catalog

2.1.1 Galactic O-Star Catalog

O Galactic O-Star Catalog (GOSC), é o mais amplo catálogo de estrelas O disponível, contendo em sua versão mais atual 655 estrelas com as respectivas classificações baseadas nos resultados do Galactic O-Star Spectroscopic Survey (Maíz Apellániz et al., 2011), que obtém espectros normalizados de estrelas Galácticas com resolução baixa-média (R = 2500). Os dados usados neste trabalho são referentes à versão 4.2 do GOSC.

Através da ferramenta de busca disponível no site do catálogo, foram retornadas todas as estrelas O disponíveis, omitindo do resultado da busca estrelas early-type de outros tipos, bem como estrelas late-type (HD 150898 e BD+01 3974, por exemplo). Em seguida realizamos o download de todos os respectivos FITS disponíveis, totalizando 588 arquivos. Deste ponto em diante foram removidas estrelas que apresentavam tipos espectrais como "OC" e "ON" (tais estrelas foram removidas também da amostra de Martins), estrelas cujo espectro não cobria regiões contendo linhas de interesse, bem como estrelas que também estavam presentes na segunda amostra, baseada no artigo de Martins, de modo a evitar duplicatas (nestes casos a escolha quase sempre foi excluir os espectros do GOSC, preferindo os obtidos das fontes do artigo, devido à maior resolução espectral). Mais detalhes a respeito deste artigo serão discutidos na subseção seguinte. Após este processo de corte, restaram 507 espectros oriundos do GOSC.

2.1.2 Amostra de Martins (2018)

No artigo de Martins (2018), com a finalidade de refinar critérios de classificação de estrelas O, foram selecionadas 105 estrelas com espectros de alta resolução disponíveis em várias fontes distintas. Neste artigo, utilizamos 99 destes espectros para complementar a amostra obtida por meio do GOSC. As estrelas HD 13268, HD 14633 e HD 201345 não foram incluídas por serem estrelas de tipo "ON". A estrela HD 207898 foi excluída pois não encontramos seu espectro disponível para *download*. E as estrelas HD 151804 e HD 155889 foram exluídas de nossa amostra devido à problemas na extração da informação de fluxo e comprimento de onda dos

arquivos FITS. As fontes dos espectros que utilizamos estão descritas abaixo:

- The *SOPHIE* archive[2] (Moultaka et al., 2004a). Arquivo contendo os dados do espectrógrafo échelle *SOPHIE*, montado no telescópio de 193 cm do Observatório de Haute-Provence (Bouchy et al., 2013). Ele cobre o intervalo de comprimento de onda de 3872-6943 Å com um poder de resolução de 40000. Os dados do arquivo foram reduzidos com a *pipeline* do *SOPHIE* (Bouchy et al., 2009).

- The *ELODIE* archive[3] (Moultaka et al., 2004b). *ELODIE* foi o espectrógrafo que precedeu o *SOPHIE* no telescópio de 193 cm do Observatório de Haute-Provence, cobrindo o intervalo de 3850-6800 Å com R = 42000. Mais detalhes da redução dos dados podem ser encontrados na *webpage* do instrumento.

- The CFHT Science Archive at the Canadian Astronomical Data Center[4]. Deste arquivo foram utilizados dados do espectro-polarímetro *ESPaDOnS*. O instrumento cobre o intervalo de 3700-10500 Å com R = 68000. A redução de dados foi realizada através do software Libre Esprit (Donati et al., 1997).

- O arquivo de espectros *NARVAL* da PolarBase[5] (Petit et al., 2014). Montado no Telescópio Bernard Lyot, no Observatório Pic du Midi, o espectro-polarímetro *NARVAL* é um instrumento gêmeo do *ESPaDOnS*. A redução de seus dados foi feita de modo similar à redução dos dados do *ESPaDOnS*.

- O arquivo do ESO[6] de espectros do *FEROS* coletados pelo survey OWN (Barbá et al., 2010). *FEROS* é um espectrógrafo instalado no telescópio de 2,2 metros de La Silla. Tem poder de resolução de 48000 e cobre o intervalo de 3500-9200 Å. Os dados foram reduzidos usando a *pipeline FEROS* do ESO.

Como dito anteriormente, nos casos de estrelas duplicadas entre as amostras do GOSC e Martins, quase sempre foram mantidas as estrelas de Martins devido à maior resolução es-

[2] http://atlas.obs-hp.fr/sophie/
[3] http://atlas.obs-hp.fr/elodie/
[4] http://www.cadc-ccda.hia-iha.nrc-cnrc.gc.ca/en/cfht/
[5] http://polarbase.irap.omp.eu/
[6] http://archive.eso.org/cms.html

pectral. Apenas nos casos de espectros demasiadamente ruidosos optamos por manter a versão do GOSC.

Ao reunir as estrelas de ambas as amostras, fica claro que certos tipos espectrais estão sub-representados se comparados a outros tipos. De um modo geral, nota-se que as estrelas mais quentes tem menor representação do que as mais frias. Isto não é surpreendente uma vez que consideremos a IMF, mas ainda sim é uma observação válida. Na Tabela 2.1 está disposta a quantidade de estrelas observadas em cada um dos 47 tipos espectrais presentes no *dataset*.

Tabela 2.1: Quantidade de Estrelas Observadas por Tipo Espectral

Tipo Espectral	Quantidade	Tipo Espectral	Quantidade
O3I	1	O7III	10
O3V	2	O7V	44
O3.5I	5	O7.5I	10
O3.5III	2	O7.5III	15
O3.5V	5	O7.5V	34
O4I	8	O8I	10
O4III	4	O8III	11
O4V	9	O8V	40
O4.5I	3	O8.5I	6
O4.5III	8	O8.5III	16
O4.5V	6	O8.5V	22
O5I	3	O9I	13
O5III	5	O9III	23
O5V	12	O9V	13
O5.5I	2	O9.2I	7
O5.5III	3	O9.2III	16
O5.5V	14	O9.2V	7
O6I	7	O9.5I	3
O6III	11	O9.5III	31
O6V	20	O9.5V	20
O6.5I	6	O9.7I	17
O6.5III	16	O9.7III	44
O6.5V	27	O9.7V	8
O7I	7		

A fim de melhor avaliar as limitações do método k-means e de nossos dados, também utilizamos dados artificiais para ampliar o tamanho de nossa amostra até um total de 47.000 elementos, dos quais 606 são as estrelas originais, e os 46.394 restantes foram gerados artificialmente. Na seção seguinte justificaremos esse número e detalharemos brevemente o critério utilizado para sintetizar esses dados suplementares.

2.2 Dados Artificiais

Para esta tarefa, utilizamos as larguras equivalentes reais como base para a criação de larguras artificiais, dando preferência às oriundas da amostra de Martins, já que podemos associar a melhor resolução espectral à medição mais precisa das larguras equivalentes. Os dados do GOSC tiveram papel complementar, sendo utilizados apenas nos casos de ausência de estrelas de certas classes na amostra de Martins. São estas: O3I, O3.5III, O4.5I, O4.5V, O5.5I, O5.5III, O6.5I, O7III e O8.5I, totalizando 9. Os dados artificiais das 38 classes restantes foram criados utilizando apenas as larguras equivalentes de Martins como base.

Cada uma das larguras artificiais foi gerada utilizando a fórmula 2.1, onde, para uma classe existente nos dados reais, calculamos o valor médio (μ) das larguras pertencentes a uma das *features*. No caso de alguma classe ser composta apenas por uma estrela, utilizamos o próprio valor da largura equivalente como μ, dispensando o cálculo da média. Como queremos aumentar o tamanho da amostra sem aumentar a dificuldade do k-means de distinguir as aglomerações no espaço de *features*, escolhemos arbitrariamente gerar cada uma das novas larguras utilizando a largura média da classe encontrada previamente com um desvio (σ) de no máximo 10%, de modo a gerar "estrelas" bastante similares. Não testamos outras porcentagens, mas 10% mostrou-se suficiente para nossos propósitos. Assim, o desvio foi multiplicado por um valor aleatório no intervalo $[-1, 1]$ e somado à média, resultando na largura artificial. Ao efetuar estes passos para as 7 *features* (que serão apresentadas no Capítulo 3), geramos uma nova estrela da classe escolhida. Esses passos foram realizados de modo que cada uma das 47 classes contabilizasse 1.000 elementos[7], totalizando 47.000 estrelas (amostra artificial) no *dataset*. A escolha de 1.000 elementos por classe foi arbitrária, baseada na necessidade de atingir uma quantidade satisfatória de elementos que eliminasse problemas relacionados a pouco volume de dados e à subamostragem de certos tipos espectrais.

$$W_\lambda(artificial) = \mu + random[-1, 1] \cdot \sigma \tag{2.1}$$

[7]Para os fins do trabalho escolhemos 1.000 estrelas por tipo espectral, entretanto lembramos que essa distribuição igualitária de estrelas não representa a natureza por causa da IMF.

Na seção seguinte veremos detalhes a respeito do método k-means, algoritmo usado na tarefa de classificação tanto do *dataset* de 606 estrelas quanto do de 47.000 (amostra artificial).

2.3 Classificação

Como visto anteriormente, a classificação correta de todas as estrelas analisadas é conhecida a priori, de modo que a classificação com aprendizado de máquina se baseará numa análise da homogeneidade dos grupos de estrelas identificados com a finalidade de testar a eficácia do algoritmo não supervisionado escolhido (k-means) na análise de nosso *dataset*. Ressaltamos também que os subtipos espectrais OII e OIV foram renomeados como OIII para adequá-los ao critério de classificação usado neste trabalho, conforme foi explicado na subseção 1.2.2. As *features* dos dados observados foram calculadas utilizando as larguras equivalentes obtidas por meio da medição com o código Python (ver Seção 2.1), e as *features* artificiais foram calculadas utilizando as larguras artificiais obtidas conforme explicado na Seção 2.2. Tabelas de estrelas observadas e artificiais com suas respectivas *features* foram colocadas no Apêndice A para melhor visualização. É importante dizer que todos os algoritmos utilizados neste trabalho foram escritos na linguagem Python com o auxílio das bibliotecas scikit-learn v0.23.2 (Pedregosa et al., 2011), pandas v1.1.3 (McKinney, 2010; Reback et al., 2020), NumPy v1.19.2 (Harris et al., 2020), Matplotlib v3.3.2 (Hunter, 2007) e SciPy v1.5.2 (Virtanen et al., 2020) em diferentes etapas. A seguir, descreveremos em detalhes o método não supervisionado escolhido para a análise.

2.3.1 k-means

O método k-means é um tipo de algoritmo não supervisionado que realiza a classificação dos dados por meio de um processo chamado *clustering*, que envolve separar os dados em grupos distintos com base em algum critério pré-estabelecido. No caso do método k-means, este critério está relacionado ao valor do hiperparâmetro k, que corresponde ao número de *clusters* que se supõe que estejam presentes nos dados a serem analisados e que, portanto, será igual ao

número de classes encontradas pelo algoritmo ao término do processo de classificação. Para fins explicativos, na Figura 2.1 estão representadas algumas etapas do processo de classificação de dados meramente ilustrativos usando o método k-means.

Após fornecer os dados ao algoritmo, k pontos são definidos aleatoriamente no espaço N-dimensional, onde os k pontos representam os centroides de cada *cluster* e N corresponde ao número de *features* definidas anteriormente. No caso da Figura 2.1, com a finalidade de criar uma representação gráfica bidimensional, apenas duas *features* arbitrárias foram utilizadas. Em seguida calcula-se a distância de cada ponto da amostra até cada centroide, para que assim, após atribuir cada ponto ao seu centroide mais próximo, a identificação dos clusters *clusters* seja propriamente iniciada.

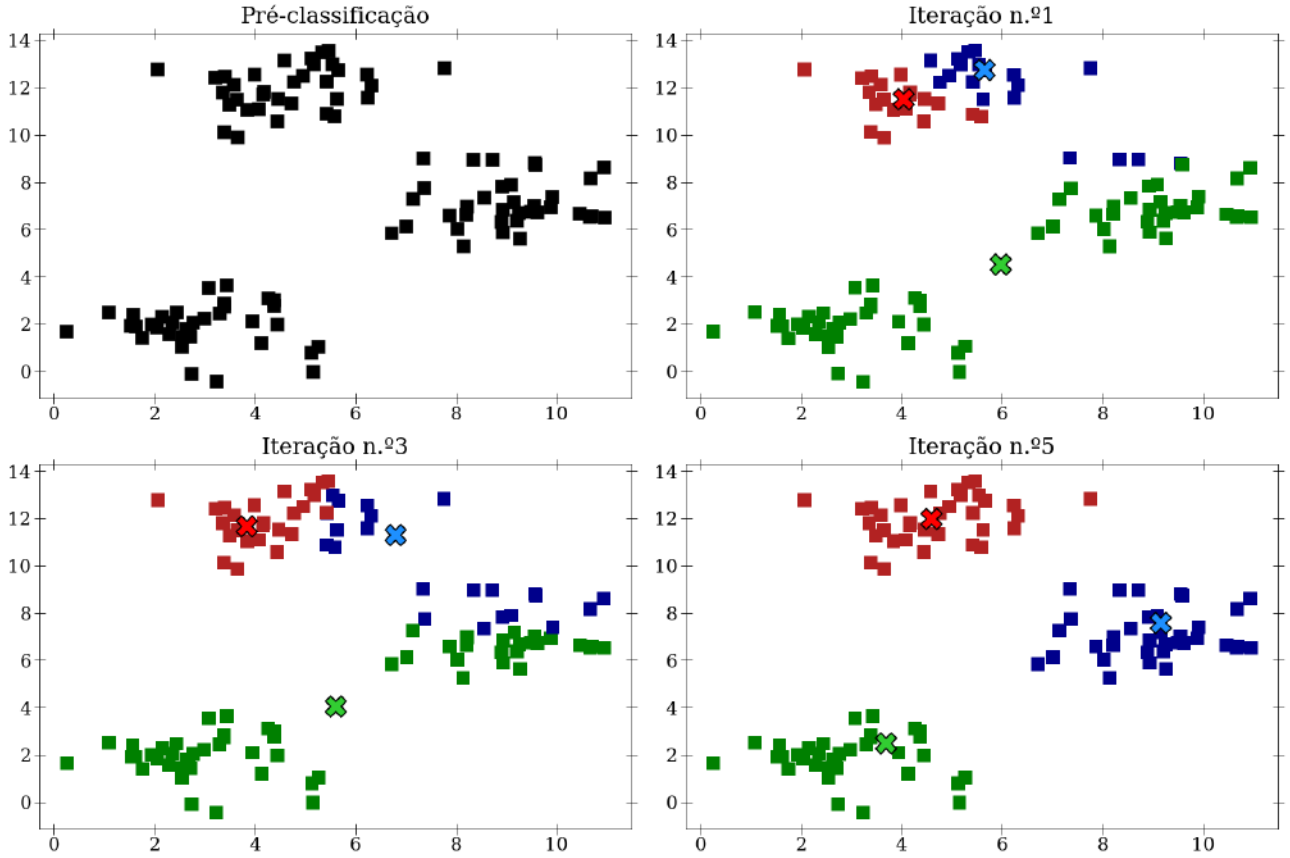


Figura 2.1: Etapas da identificação bem-sucedida de *clusters* em dados ilustrativos – O gráfico acima ilustra a distribuição dos pontos de uma amostra qualquer ao longo de dois eixos (associados a duas *features*) antes da classificação k-means, bem como na primeira, terceira e quinta etapas de sua iteração. Os marcadores "X" coloridos identificam as coordenadas dos centroides de cada um dos três *clusters*, os quadrados coloridos identificam seus respectivos pontos, e os quadrados pretos identificam pontos não classificados.

Na etapa seguinte, cada centroide tem sua posição alterada para a posição média dos pontos pertencentes a seus *clusters*, então mais uma vez calcula-se a distância entre todos os pontos da amostra e todos os centroides e atribui-se cada ponto ao seu centroide mais próximo, concluindo mais uma iteração do algoritmo. Esse processo se repete até que os centroides não alterem mais sua posição de modo significativo, ou seja, até que a soma das distâncias euclidianas entre cada ponto e seu respectivo centroide seja minimizada o máximo possível. Neste momento afirma-se que o algoritmo convergiu para uma solução, ou mínimo local. Naturalmente, pode existir mais de uma solução para a amostra em questão e nem sempre a encontrada será a mais apropriada. A capacidade do algoritmo de convergir para a solução ideal depende da distribuição dos pontos da amostra (e portanto, das *features* usadas na classificação) e também da posição inicial dos centroides. Além disso, ao alterar o valor de k, altera-se também o universo de soluções possíveis para o problema. Na Figura 2.2, está representado um caso em que os pontos da amostra estão distribuídos com maior espalhamento, tornando mais difícil para o algoritmo identificar de maneira unívoca os *clusters* existentes mesmo após 100 iterações.

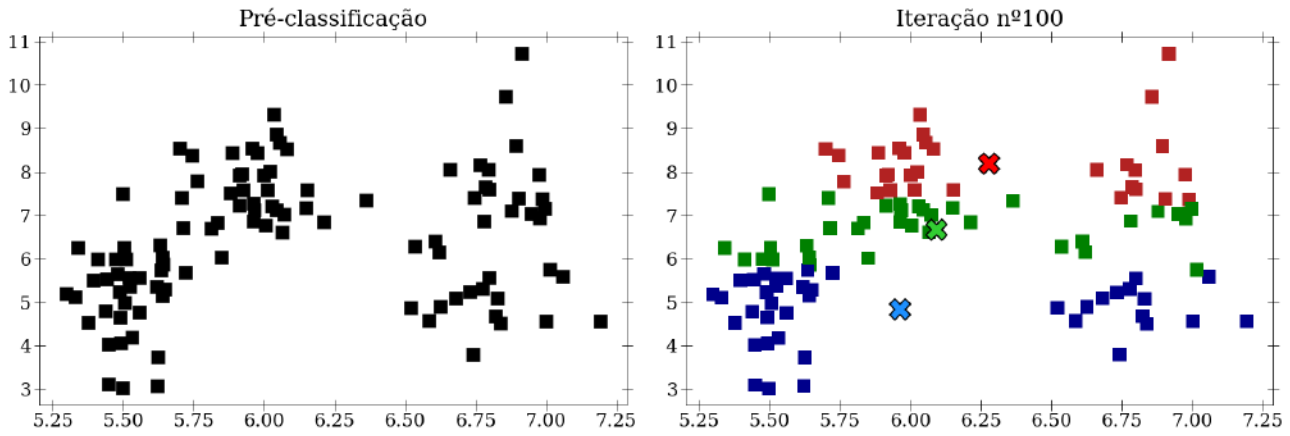


Figura 2.2: Identificação insatisfatória de *clusters* em dados ilustrativos – O gráfico acima ilustra a distribuição dos pontos de uma amostra qualquer ao longo de dois eixos (associados a duas *features*) antes da classificação k-means e em sua centésima iteração. Os marcadores cumprem a mesma função que na Figura 2.1.

Ao contrário dos exemplos anteriores, é normal que não se conheça o valor a ser atribuído ao hiperparâmetro k e, portanto, métodos para estimá-lo costumam ser utilizados. Neste trabalho escolhemos o método conhecido como silhouette score para este fim.

Silhouette Score

Para um dado valor de k, o método procura avaliar a qualidade da separação entre os *clusters* utilizando os parâmetros da fórmula 2.2, onde a_i é a distância média do ponto i de um *cluster* até todos os outros pontos deste mesmo *cluster*, b_i é a distância média do mesmo ponto i até todos os pontos do *cluster* vizinho mais próximo, $max(b_i, a_i)$ representa o maior valor entre a_i e b_i, e S_i é o *score* do ponto i. Após calcular o valor de todos os pontos da amostra, tira-se a média total dos *scores*, que é o resultado final.

$$S_i = \frac{b_i - a_i}{max(b_i, a_i)} \tag{2.2}$$

O quociente poderá ter valores entre 1 e -1, onde 1 é o melhor resultado possível, indicando uma separação clara entre os *clusters*, e -1 indica que pontos podem ter sido atribuídos ao *cluster* errado. Valores próximos a 0 indicam que os pontos estão próximos à fronteira de separação entre *clusters*, ou seja, provavelmente há sobreposição entre eles. Se o processo for repetido para diversos valores de k, é possível encontrar o valor que, idealmente, fornecerá a melhor identificação dos *clusters* presentes nos dados da amostra. Uma vez decidido o valor de k, o próprio método silhouette pode ser usado como avaliador do resultado da classificação, uma vez que podemos comparar o *score* dos elementos pertencentes a cada *cluster* com o *score* médio para o valor de k escolhido.

Uma limitação deste método está relacionada com a distribuição dos pontos no espaço de *features* em casos pouco ideais. Se os pontos da amostra utilizada não apresentarem uma separação muito clara, o método silhouette identificará poucos *clusters*, que por sua vez serão povoados por pontos pertencentes a classes distintas. O resultado disso é que por não ter conhecimento das *labels* corretas, o método poderá dar um *score* alto mesmo quando a identifição dos *clusters* não for satisfatória, levando a escolha de um valor de k que não corresponde à quantidade real de classes existentes na amostra. Tal limitação foi encontrada neste trabalho, e discutiremos mais a seu respeito no Capítulo 3, onde relataremos os resultados da aplicação dos métodos descritos acima nos dados que utilizamos.

Capítulo 3

Resultados

3.1 Aplicação do Método k-means

3.1.1 Escolha de Features

As *features* que utilizamos em nossa classificação são as larguras equivalentes e razões de larguras equivalentes discutidas no Capítulo 1. Como um dos principais focos deste trabalho é testar a capacidade de nosso algoritmo de identificar e agrupar estrelas levando em conta tanto os subtipos espectrais como as classes de luminosidade, as classificações utilizaram todas as *features* simultaneamente, totalizando sete. Entretanto, testes também foram realizados utilizando separadamente as *features* relevantes para a identificação de subtipos espectrais e as relevantes para as classes de luminosidade com a finalidade de analisar a influência individual destes aspectos na eficácia do algoritmo.

Conforme visto em Martins (2018), para determinação do subtipo espectral as features são:
W_λ[He I λ4471]/W_λ[He II λ4542]
W_λ[He I λ4144]/W_λ[He II λ4200]
W_λ[He I λ4388]/W_λ[He II λ4542]
W_λ[Si III λ4552]/W_λ[He II λ4542]

E para a determinação das classes de luminosidade usamos:

W_λ[He II λ4686]

W_λ[He II λ4686]/W_λ[He I λ4713]

W_λ[Si IV λ4089]/W_λ[He I λ4026]

Além das *features*, também foi necessário definir valores para o hiperparâmetro k em cada um dos testes realizados. A seguir detalharemos o processo de escolha desse valor com base no silhouette score.

3.1.2 Definição do Hiperparâmetro k

Tendo em mente que neste projeto utilizamos dados de estrelas já classificadas, sabemos que para o caso em que o objetivo é classificar simultaneamente em subtipos espectrais e classes de luminosidade, o número total de classes é igual a 47, e que portanto, numa situação de classificação ideal, deveríamos encontrar $k = 47$ como resultado da estimação usando o silhouette score. Mas como veremos a seguir, isso não ocorreu em nossos testes.

Para a realização dos testes com o silhouette score em específico, utilizamos a função `sklearn.metrics.silhouette_score` da biblioteca Scikit-learn com os parâmetros *default*. Essa função retorna o score médio calculado a partir de todos os pontos da amostra após a classificação, isto é, com base nas features e na previsão feita pelo algoritmo k-means para dado valor de k, o score de cada estrela é calculado, e então obtém-se a média. O teste foi realizado para valores de k no intervalo $[2, 57]$ com os respectivos *scores* plotados no gráfico da Figura 3.1. O gráfico revela que o *score* mais próximo de 1 é obtido quando $k = 2$, e que para valores em torno de $k = 47$ o *score* é apenas um pouco melhor que 0.2. Devido às condições iniciais aleatórias do teste, várias iterações foram realizadas, entretanto houve pouca variação nos resultados. Também realizamos o mesmo teste, cujo gráfico está na Figura 3.2, com o *dataset* contendo 47.000 estrelas (amostra artificial) para efeito de comparação.

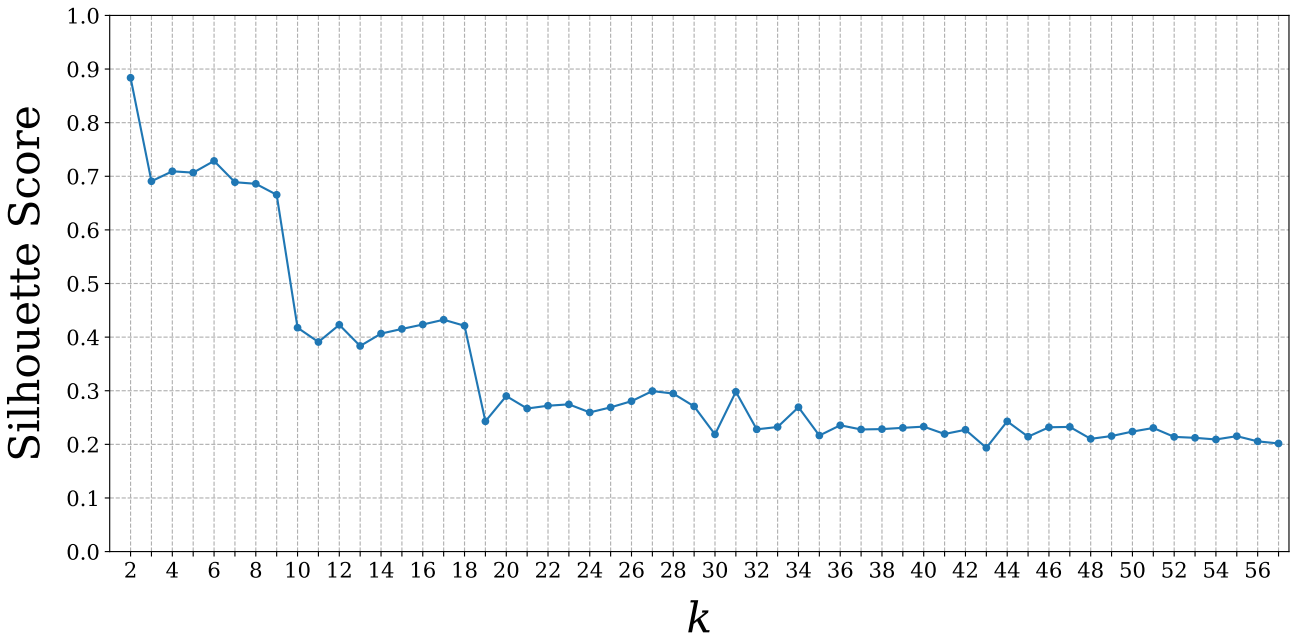


Figura 3.1: Silhouette score médio em função de k da amostra de 606 estrelas – Os pontos azuis representam os *scores* médios, cujos valores estão dispostos no eixo ordenado, e os respectivos valores de k estão dispostos no eixo das abscissas.

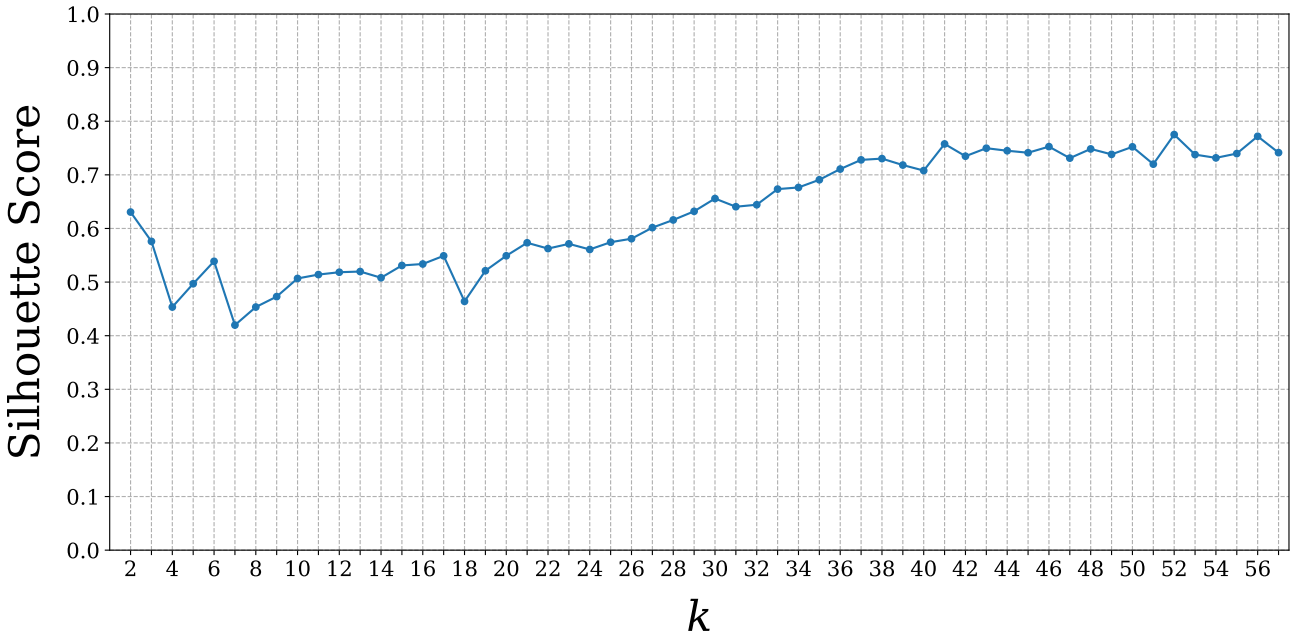
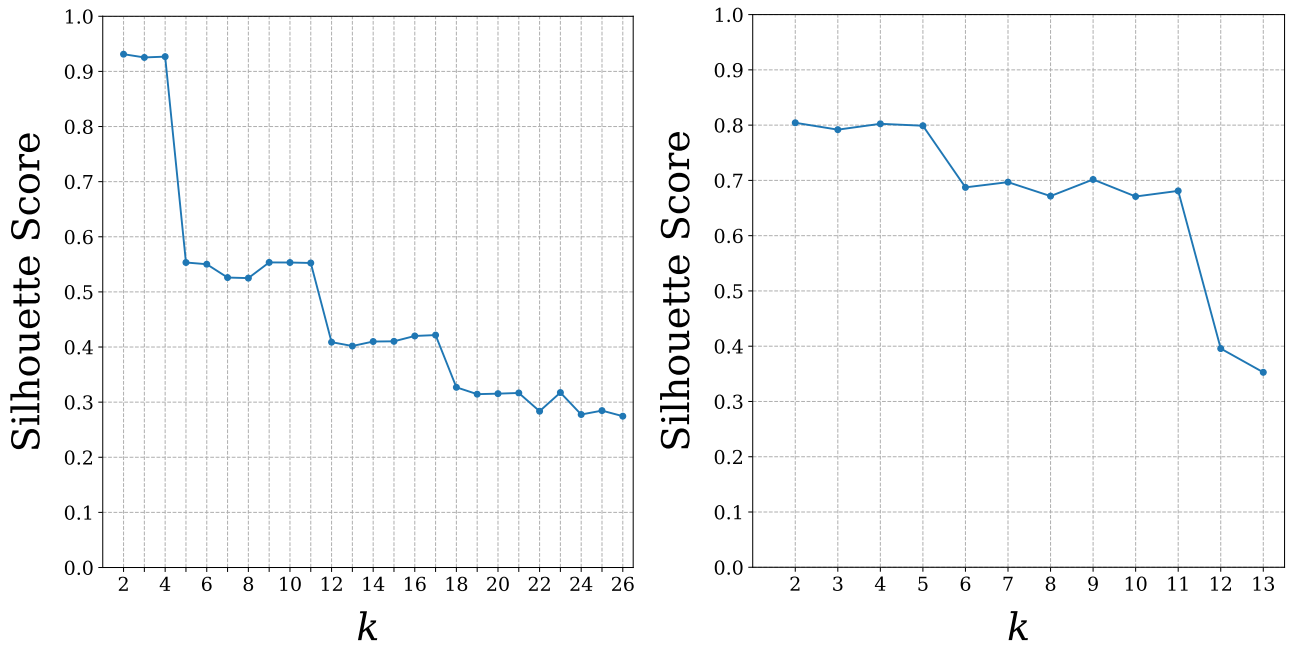


Figura 3.2: Silhouette score médio em função de k da amostra de 47.000 estrelas (amostra artificial).

No segundo caso há uma melhora substancial no resultado do teste, com o resultado para $k = 47$ atingindo um *score* um pouco acima de 0.7, tornando aceitável a escolha deste valor para o hiperparâmetro k, ao contrário do primeiro caso, onde, naturalmente, é fácil perceber

que o valor de $k = 2$ não é apropriado, uma vez que classificar 606 estrelas de 47 classes distintas em apenas 2 *clusters* resultaria em agrupamentos totalmente heterogêneos, dos quais seria impossível obter alguma conclusão razoável a respeito dos tipos espectrais. A mesma conclusão é válida para quaisquer valores de k que estejam muito distantes de 47.

Como mencionado em 3.1.1, também calculamos o silhouette score em dois casos de menor dimensionalidade através da seleção específica das *features* relevantes para a identificação de subtipos espectrais, e então das *features* relevantes para a identificação de classes de luminosidade. No primeiro caso, temos $k = 16$ como valor correto. Já no caso em que o foco são apenas as classes de luminosidade, $k = 3$ é correto. Para o *dataset* de 606 estrelas, obtivemos os resultados dispostos na Figura 3.3. Foram testados valores de k no intervalo $[2, 26]$ para o caso da classificação apenas em subtipos espectrais (3.3a), e $[2, 13]$ para o caso da classificação apenas em classes de luminosidade (3.3b).



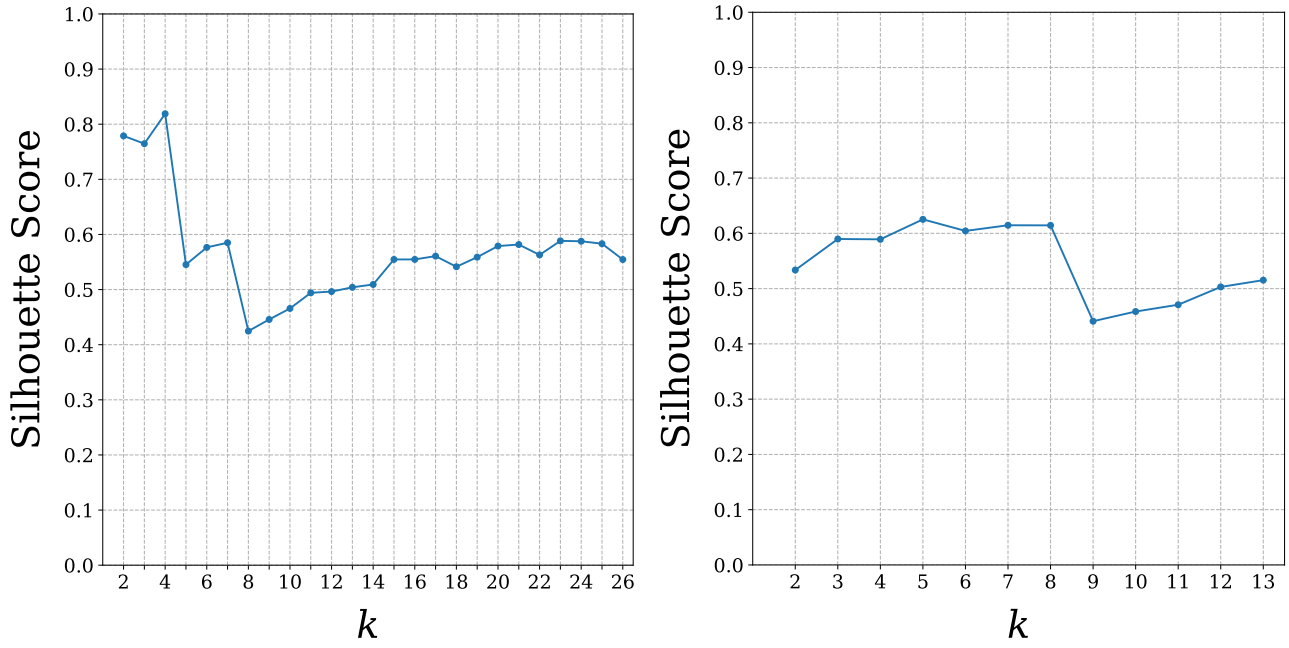
(a) Subtipos espectrais

(b) Classes de luminosidade

Figura 3.3: Silhouette score médio em função de k da amostra de 606 estrelas para os casos de menor dimensionalidade – Os eixos e pontos têm o mesmo significado que os da Figura 3.1.

Na Figura 3.3a, temos que $k = 16$ não atinge um *score* de valor satisfatoriamente alto, enquanto que na Figura 3.3b o resultado atingiu nossas expectativas, com um score alto para $k = 3$.

No teste realizado com o *dataset* de 47.000 estrelas (amostra artificial), testamos valores de k nos mesmos intervalos e, como mostra a Figura 3.4, houve uma pequena melhora na classificação em subtipos espectrais, com o *score* de $k = 16$ subindo de valores próximos a 0.4 para valores próximos de 0.55 (gráfico 3.4a). Entretanto, houve piora na classificação em classes de luminosidade, com o *score* de $k = 3$ caindo de 0.8 para valores próximos de 0.6 (gráfico 3.4b).



(a) Subtipos espectrais

(b) Classes de luminosidade

Figura 3.4: Silhouette score médio em função de k da amostra de 47.000 estrelas (amostra artificial) para os casos de menor dimensionalidade.

Levando em conta os valores corretos de k, os resultados sugerem que as *features* relacionadas aos subtipos espectrais são ao menos um dos motivos por trás da incapacidade de se determinar um valor razoável para k no teste da Figura 3.1. Além disso, a adição de dados artificiais à amostra parece influenciar positivamente a classificação em subtipos espectrais, mas a influência na classificação em classes de luminosidade parece ser negativa.

Considerando os resultados obtidos nesta subseção juntamente com o fato de que calcular o silhouette *score* médio para vários valores de k fornece apenas uma sugestão de qual valor escolher e não é uma decisão final (ver silhouette score na Subseção 2.3.1), optamos por realizar a classificação utilizando sempre os valores de k correspondentes ao número correto de *clusters*, comparando o desempenho ao classificar os *datasets* de 606 e 47.000 elementos (amostra artificial) tanto no caso em que $k = 47$, quanto nos casos em que $k = 16$ e $k = 3$.

3.1.3 Clusters Obtidos

Nesta seção mostraremos os resultados da classificação em *clusters* feita pelo algoritmo k-means para os seis testes discutidos na seção anterior. Para realizar a classificação, utilizamos a função `sklearn.cluster.KMeans` da biblioteca Scikit-learn com os parâmetros `n_init=100` e `n_clusters=k`, onde `k` é o número de clusters a ser encontrado. No Apêndice B estão disponíveis mais detalhes do código usado. Nas páginas seguintes estão dispostas imagens referentes aos testes realizados, onde cada uma delas contém histogramas representando os *clusters* identificados pelo algoritmo. O eixo das abscissas mostra as classes pertencentes ao *cluster*, enquanto no eixo das ordenadas está representada a contagem de estrelas destas classes.

A Figura 3.5 mostra os 47 *clusters* resultantes da classificação do *dataset* de 606 estrelas utilizando as 7 features. O silhouette *score* médio obtido para esta configuração foi igual a 0.23, o que demonstra que os *clusters* encontrados pelo algoritmo estão em sobreposição, indicando significativa heterogeneidade de classes. Como podemos ver nos histogramas, o resultado obtido foi insatisfatório, pois não apenas confirma o que já havia sido indicado pelo silhouette *score*, mas também evidencia uma tendência do algoritmo em povoar *clusters* com uma ou poucas estrelas. Os *clusters* 6, 30, 37 e 40 apresentaram os melhores resultados, pois possuem quantidade razoável de estrelas e são compostos predominantemente pelas mesmas classes de luminosidade ou por subtipos espectrais vizinhos (ou ambos). Na Figura 3.6 temos os histogramas obtidos após a classificação com $k = 47$ e 7 *features*, mas utilizando o *dataset* de 47.000 estrelas (amostra artificial). Conforme indicado na subseção anterior, o *score* foi muito superior, atingindo o valor de 0.76, o que significa que o algoritmo identificou *clusters* com boa separação. Podemos notar nos histogramas que a maioria dos *clusters* são compostos por quase todas as estrelas de uma única classe e que a quantidade de estrelas pertencentes a outras classes é comparativamente pequena na grande maioria dos casos. O *cluster* 37 em especial contém todas as estrelas O3I e nenhuma estrela de outra classe. Vale ressaltar também que no *dataset* de 606 estrelas havia apenas uma estrela O3I, que foi incluída no *cluster* 18 da Figura 3.5, juntamente com estrelas de várias outras classes, evidenciando o efeito positivo no resultado quando aumentamos o número de elementos de classes em subamostragem.

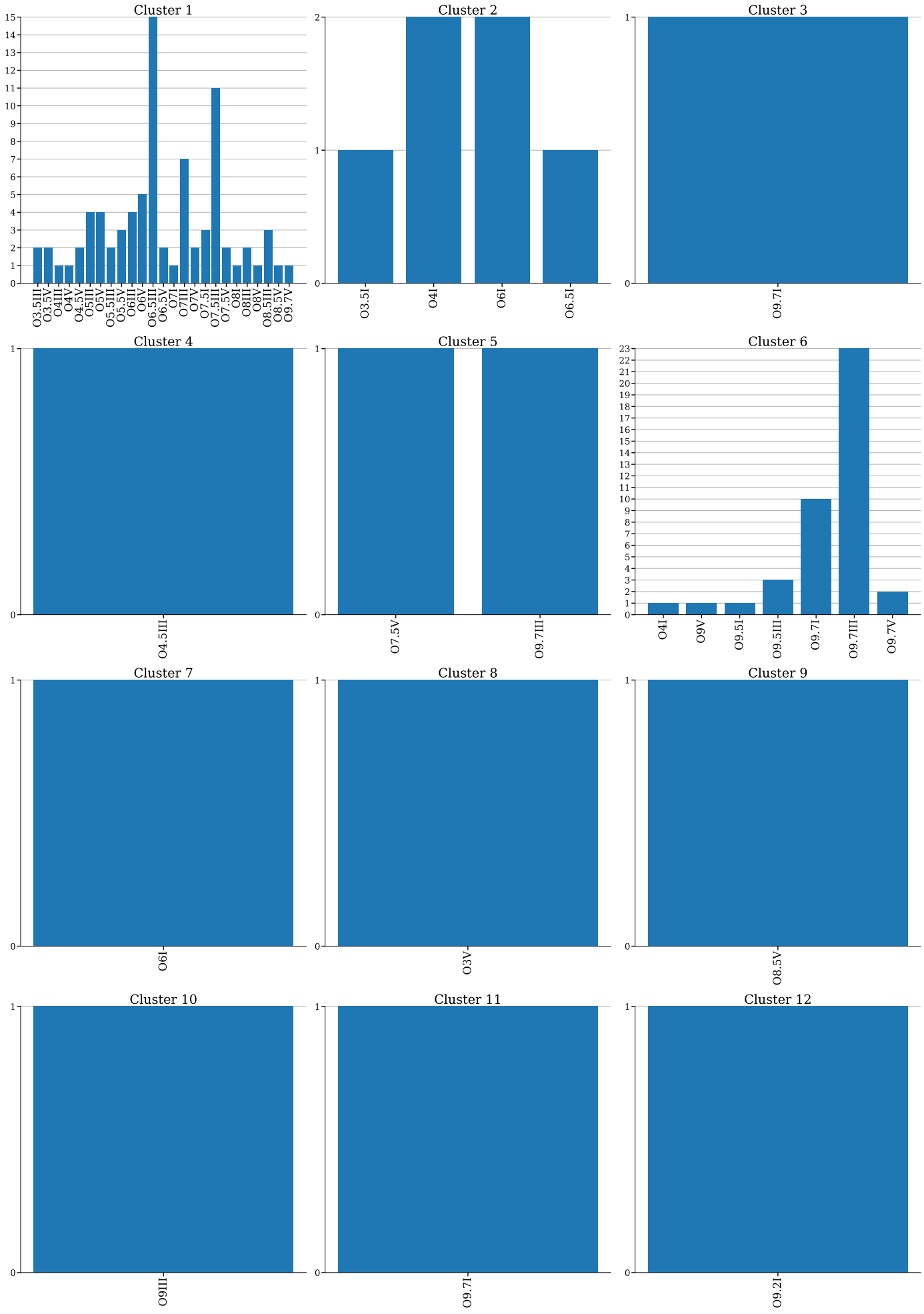


Figura 3.5: 606 estrelas, 7 *features* ($k = 47$), *score* $= 0.23$.

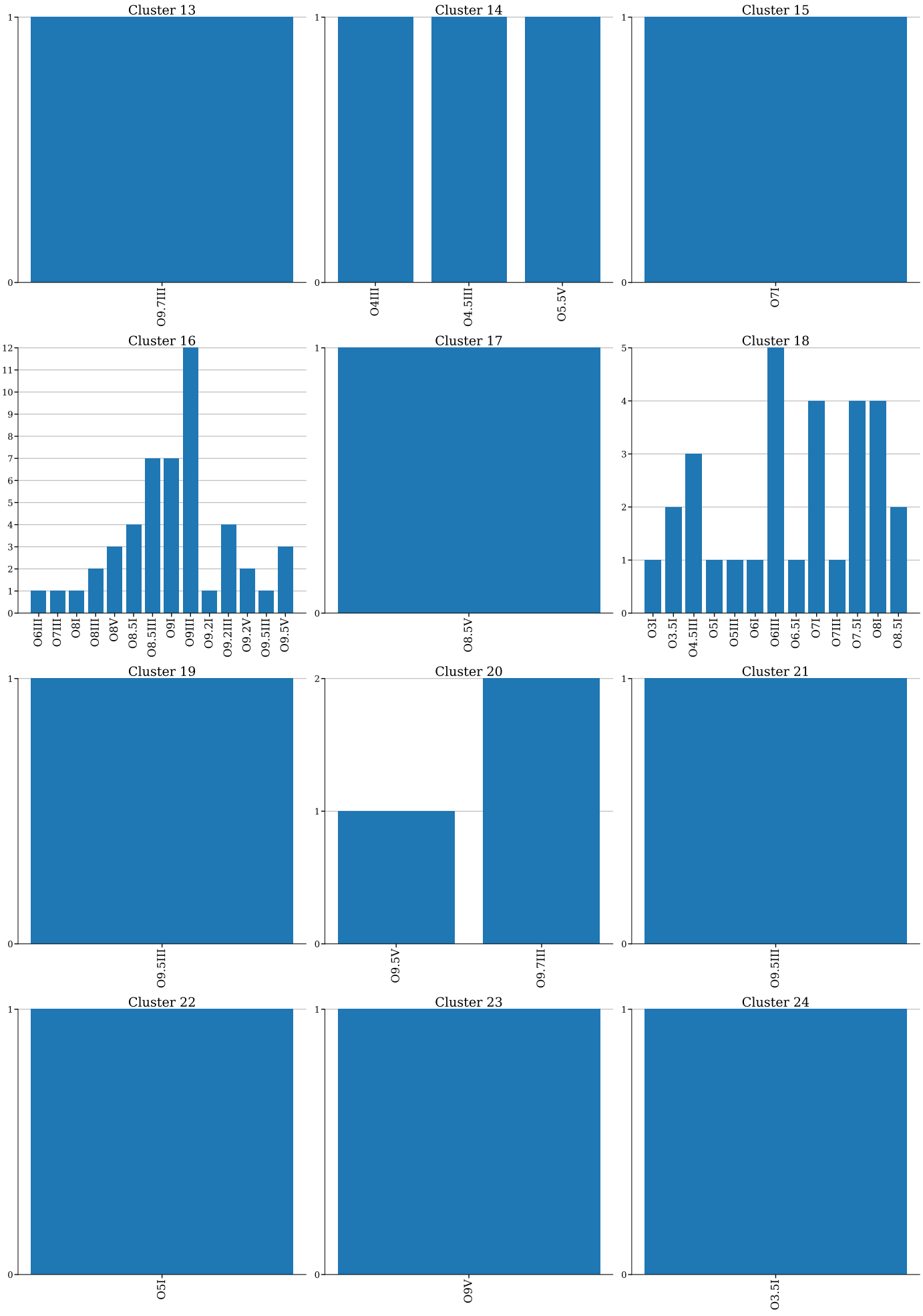


Figura 3.5: (continuação)

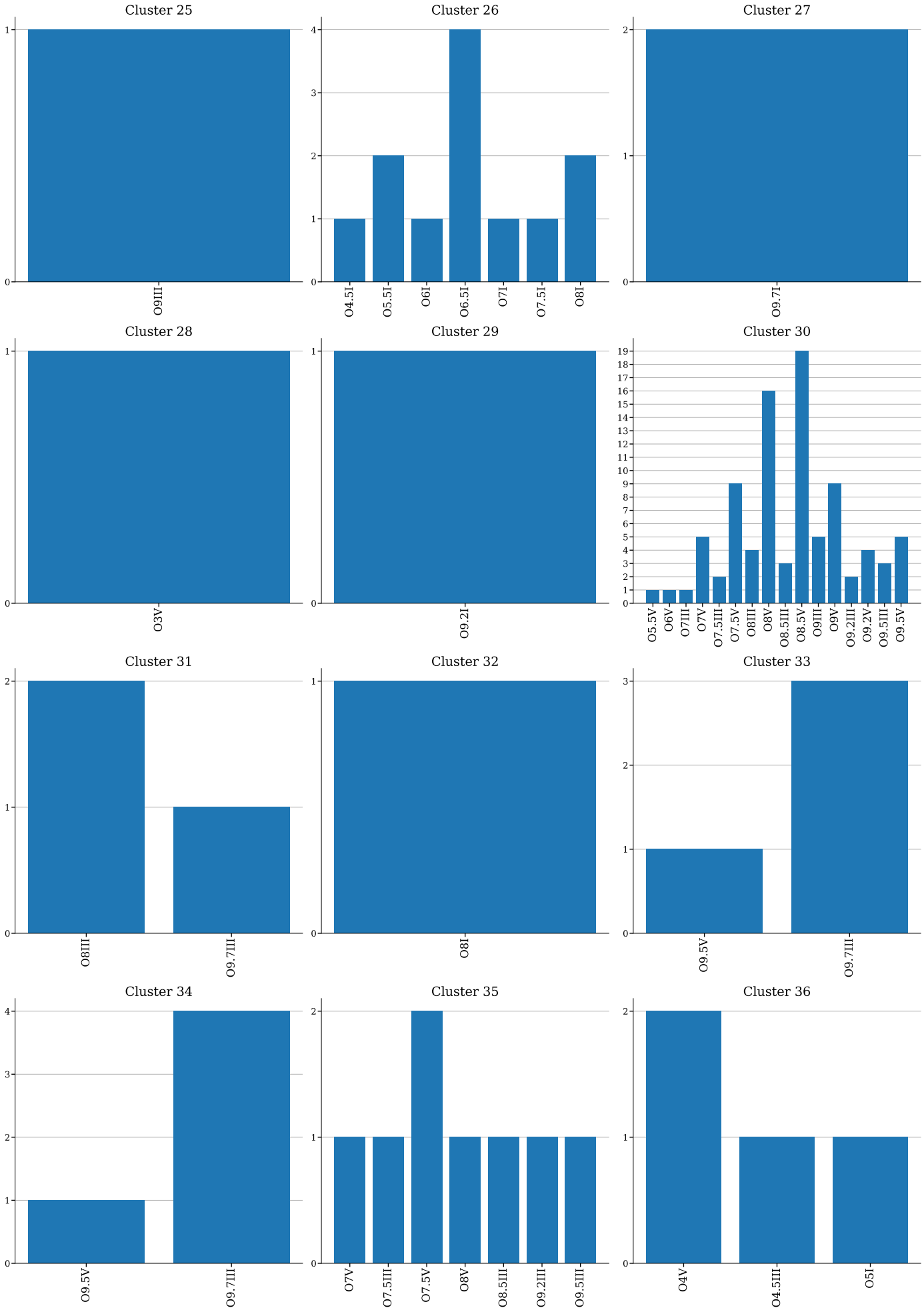


Figura 3.5: (continuação)

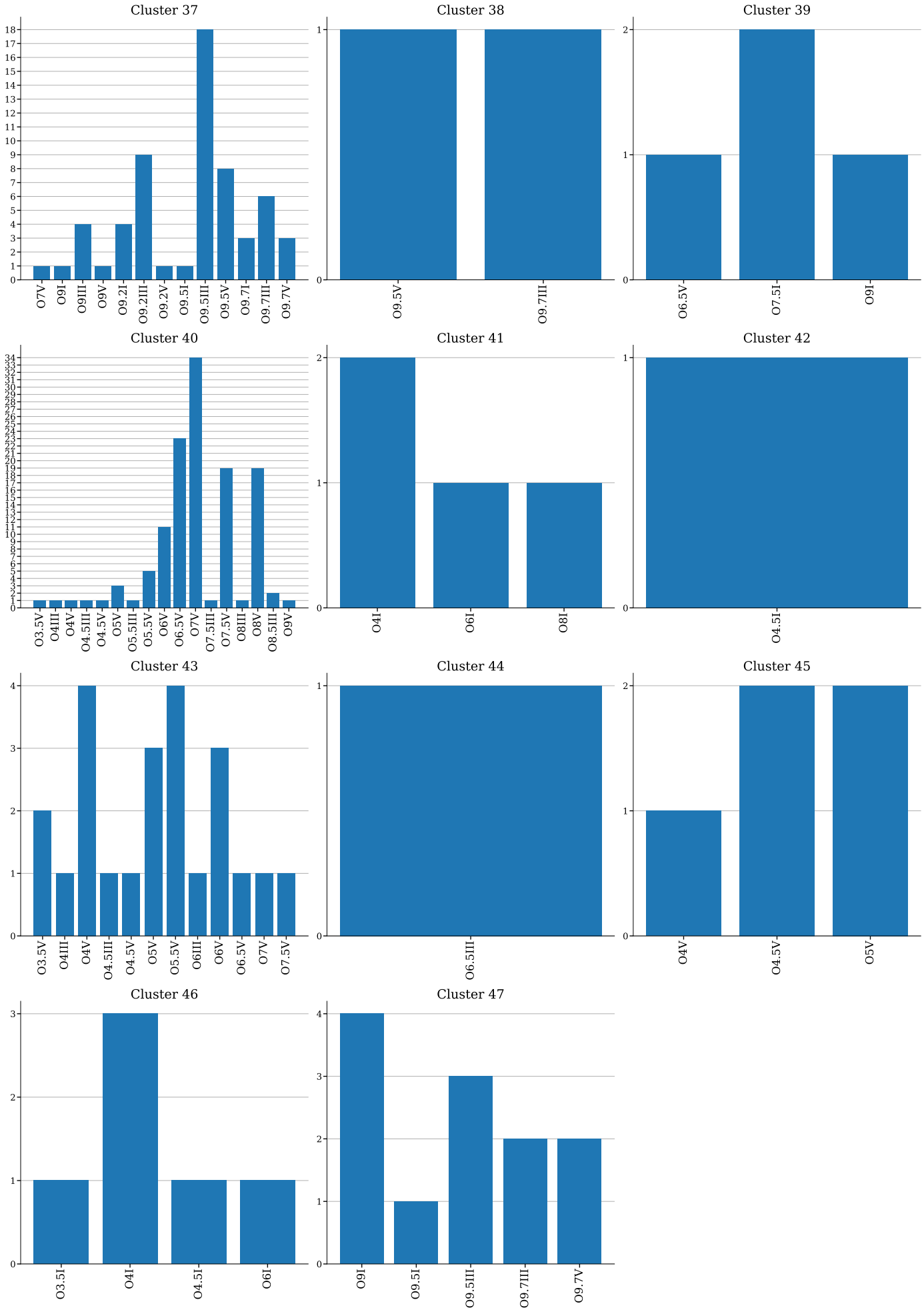


Figura 3.5: (continuação)

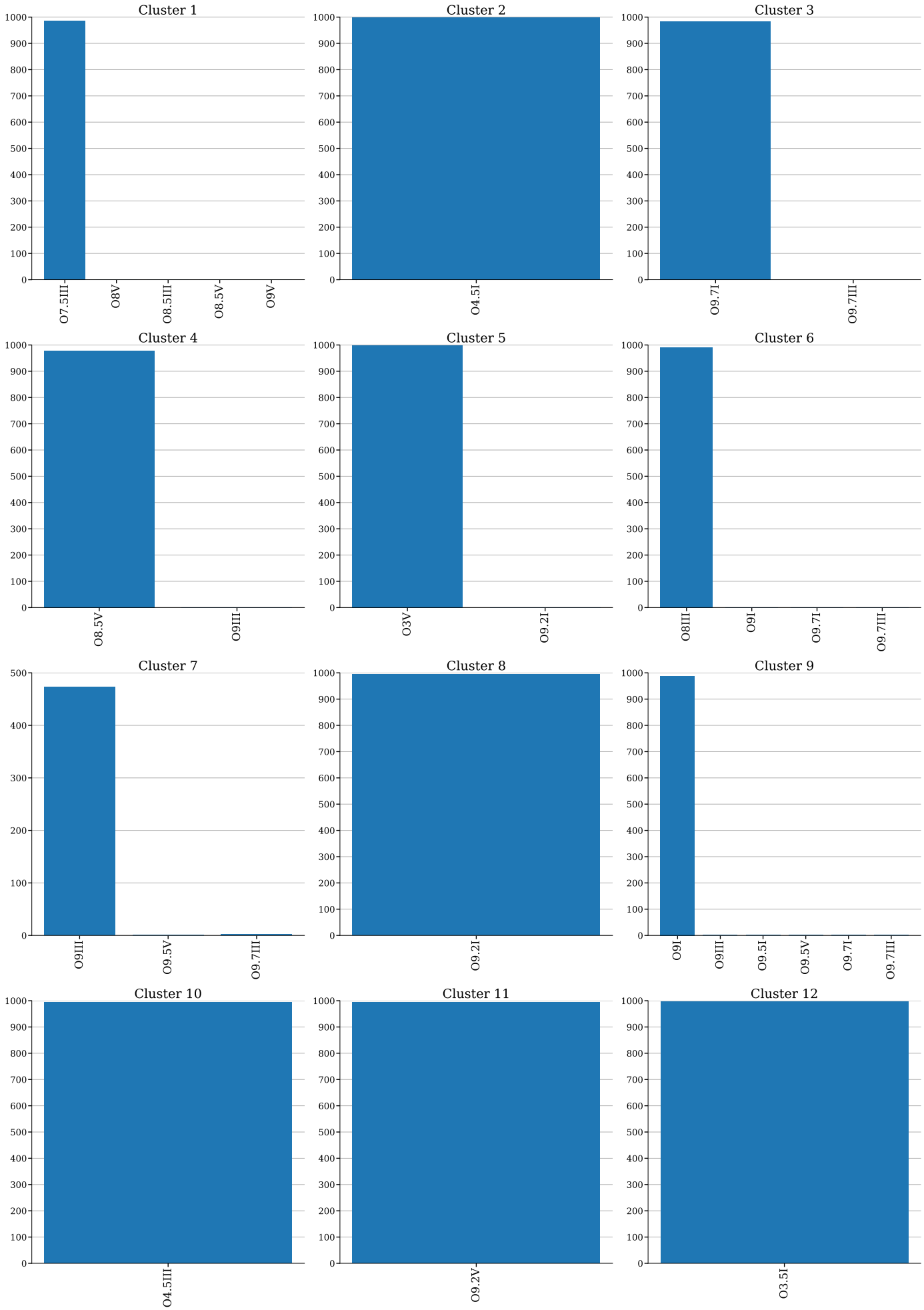


Figura 3.6: 47.000 estrelas, 7 *features* ($k = 47$), *score* $= 0.76$.

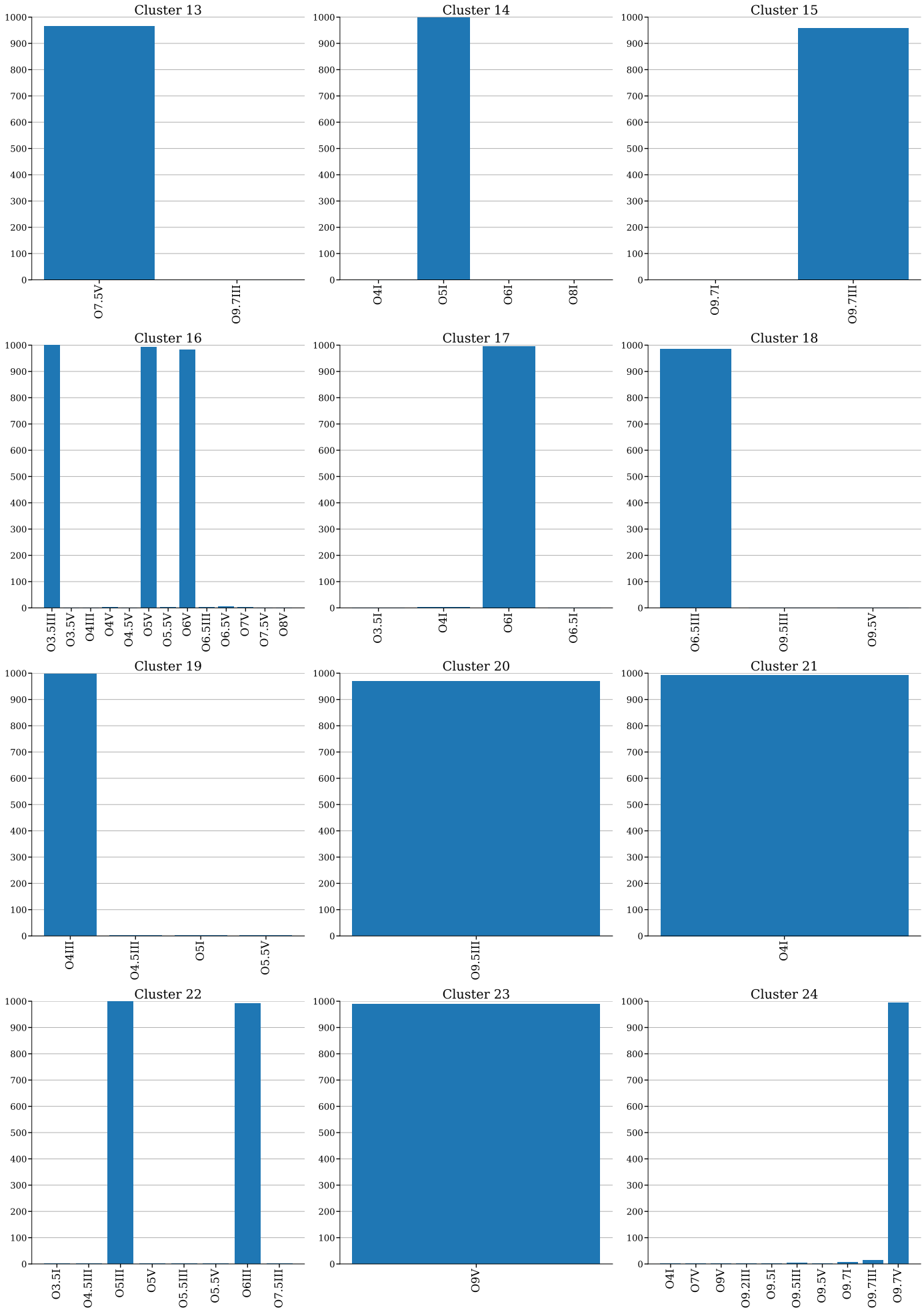


Figura 3.6: (continuação)

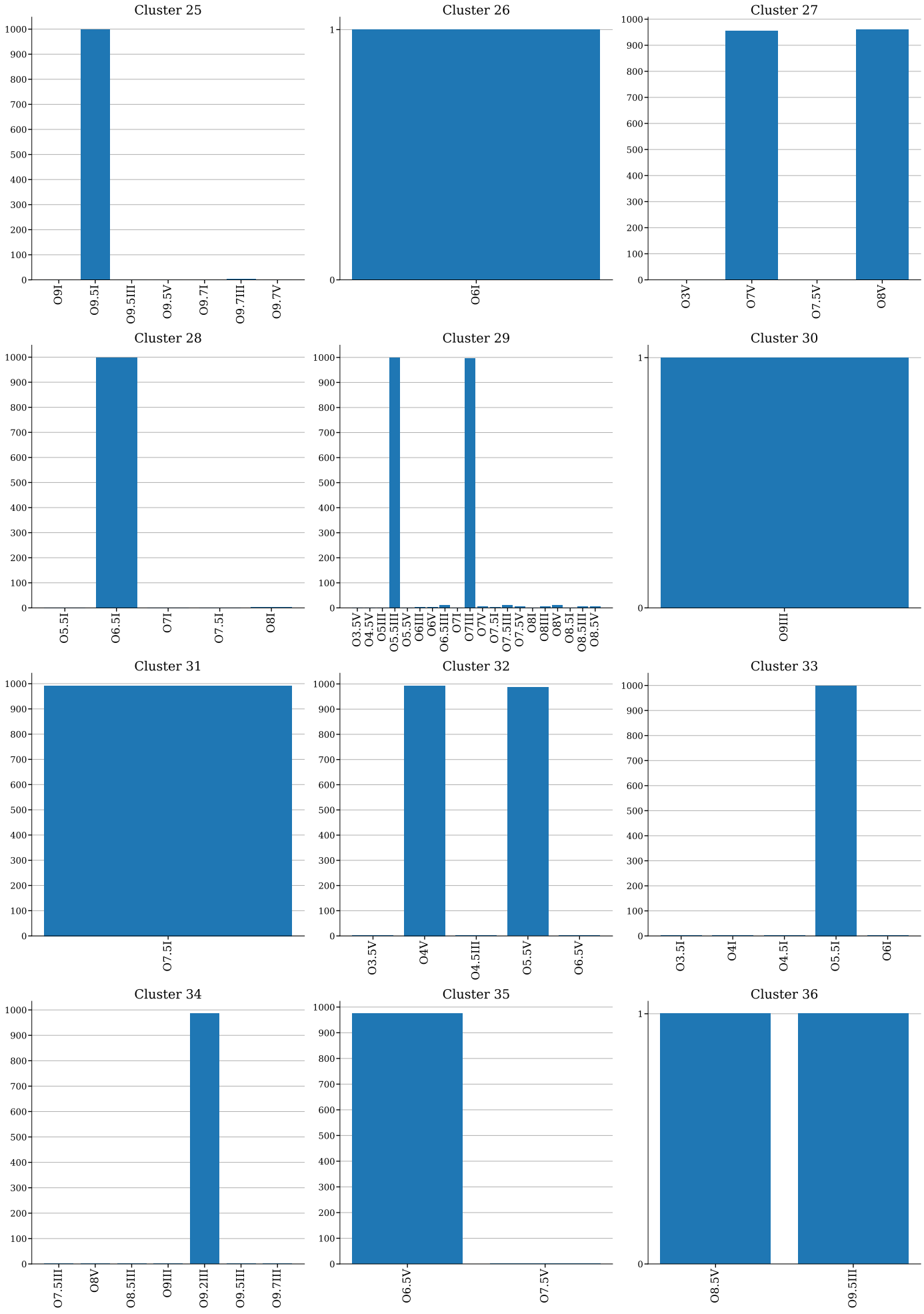


Figura 3.6: (continuação)

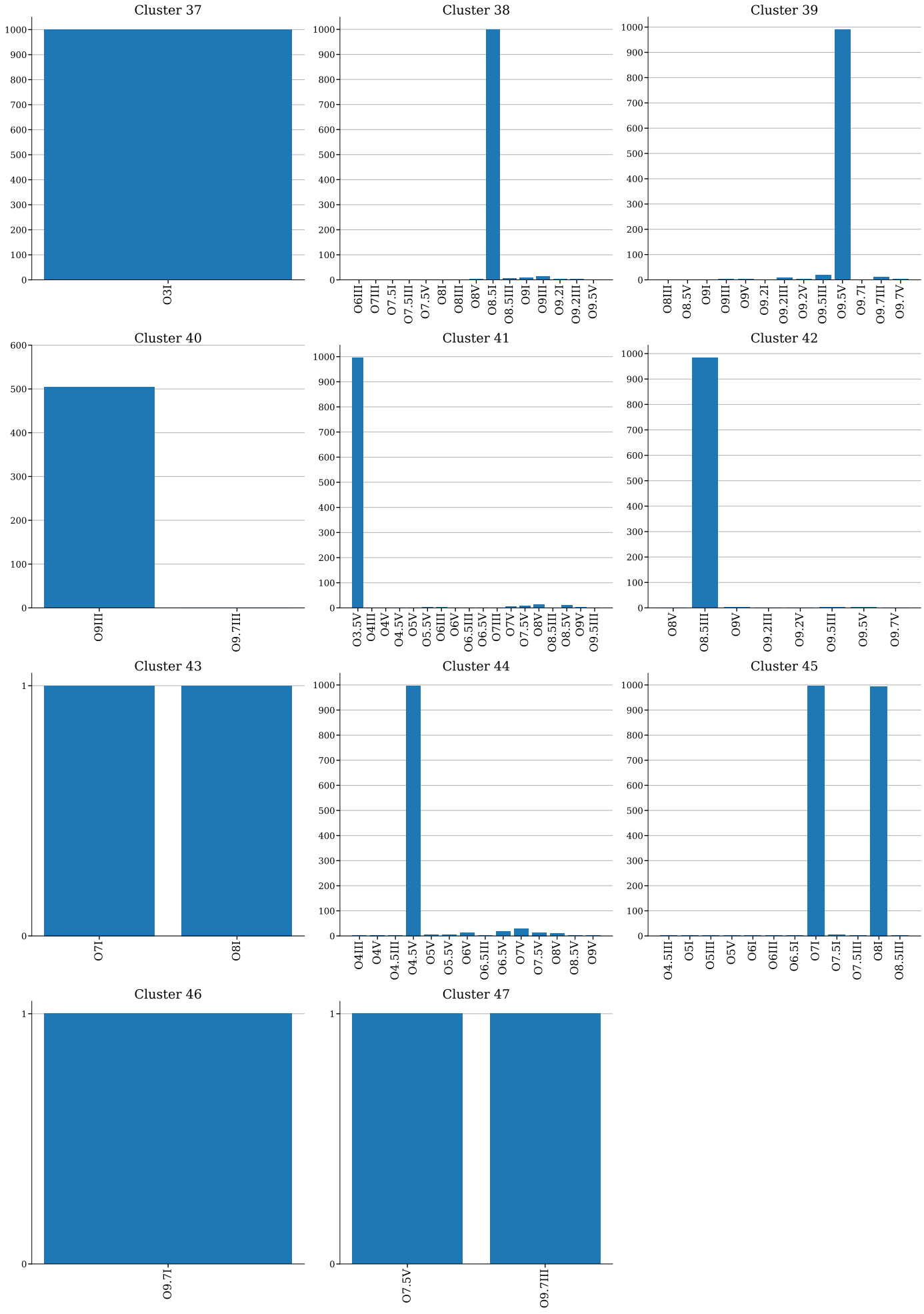


Figura 3.6: (continuação)

Como vimos nas Figuras 3.3 e 3.4, os *scores* obtidos indicam que o aumento do número de estrelas do *dataset* melhora a capacidade do algoritmo de identificar os subtipos espectrais, e afeta negativamente a capacidade de identificação das classes de luminosidade. Nas páginas seguintes, mostraremos os histogramas destas classificações de modo a tentar identificar influências do aumento do tamanho da amostra na identificação de subtipos espectrais e classes de luminosidade específicas.

A Figura 3.7 mostra os histogramas resultantes da classificação com $k = 16$ do dataset de 606 estrelas (4 *features*). Assim como em 3.3a, o *score* foi de 0.42. É possível perceber que embora a maioria dos *clusters* possuam poucos ou apenas um elemento, os subtipos inclusos nos *clusters* 4 e 12 são predominantemente vizinhos. A facilidade do algoritmo de identificar a semelhança entre os subtipos espectrais maios frios talvez se deva em parte à maior representatividade destas classes na amostra. A Figura 3.8 mostra o resultado da classificação com 3 *features* ($k = 3$) também para 606 estrelas onde o *score* foi de 0.79, conforme indicado na Figura 3.3b.

Na Figura 3.9 novamente classificamos com $k = 16$ e (4 *features*), mas desta vez com o *dataset* de 47.000 estrelas (amostra artificial). Podemos notar que embora os subtipos espectrais mais frios tenham sido melhor classificados se comparados à Figura 3.7, ainda assim há considerável mistura com subtipos quentes, que por sua vez foram ainda melhor classificados quando comparados aos resultados da classificação com 606 estrelas. Estes resultados também são compatíveis com o *score* de 0.55 obtido.

Finalmente, na Figura 3.10 temos os resultados da classificação com 3 *features* ($k = 3$) para 47.000 estrelas (amostra artificial), onde podemos notar a piora do desempenho do algoritmo se comparado ao resultado da Figura 3.8. Esta piora já havia sido indicada pelos *scores*, que caíram de 0.79 para 0.59, como visto nas figuras 3.3b e 3.4b. Enquanto na Figura 3.8 todas as estrelas de classe III e quase todas as estrelas da classe V encontram-se no *cluster* 1, na Figura 3.10 as de classe I distribuíram-se entre os 3 *clusters*, e uma proporção maior das estrelas de classe V foi separada em clusters diferentes.

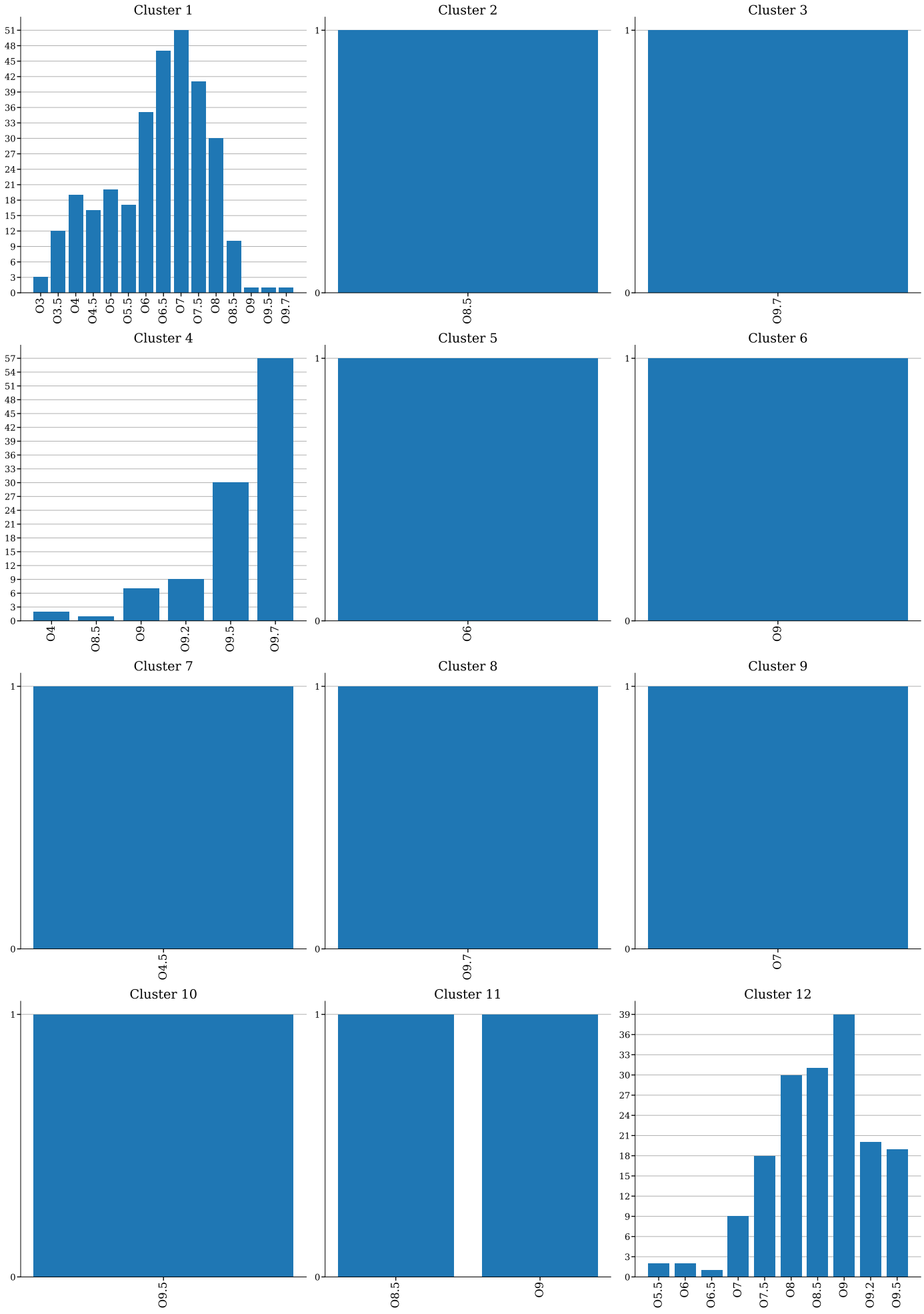


Figura 3.7: 606 estrelas, 4 *features* ($k = 16$), *score* $= 0.42$

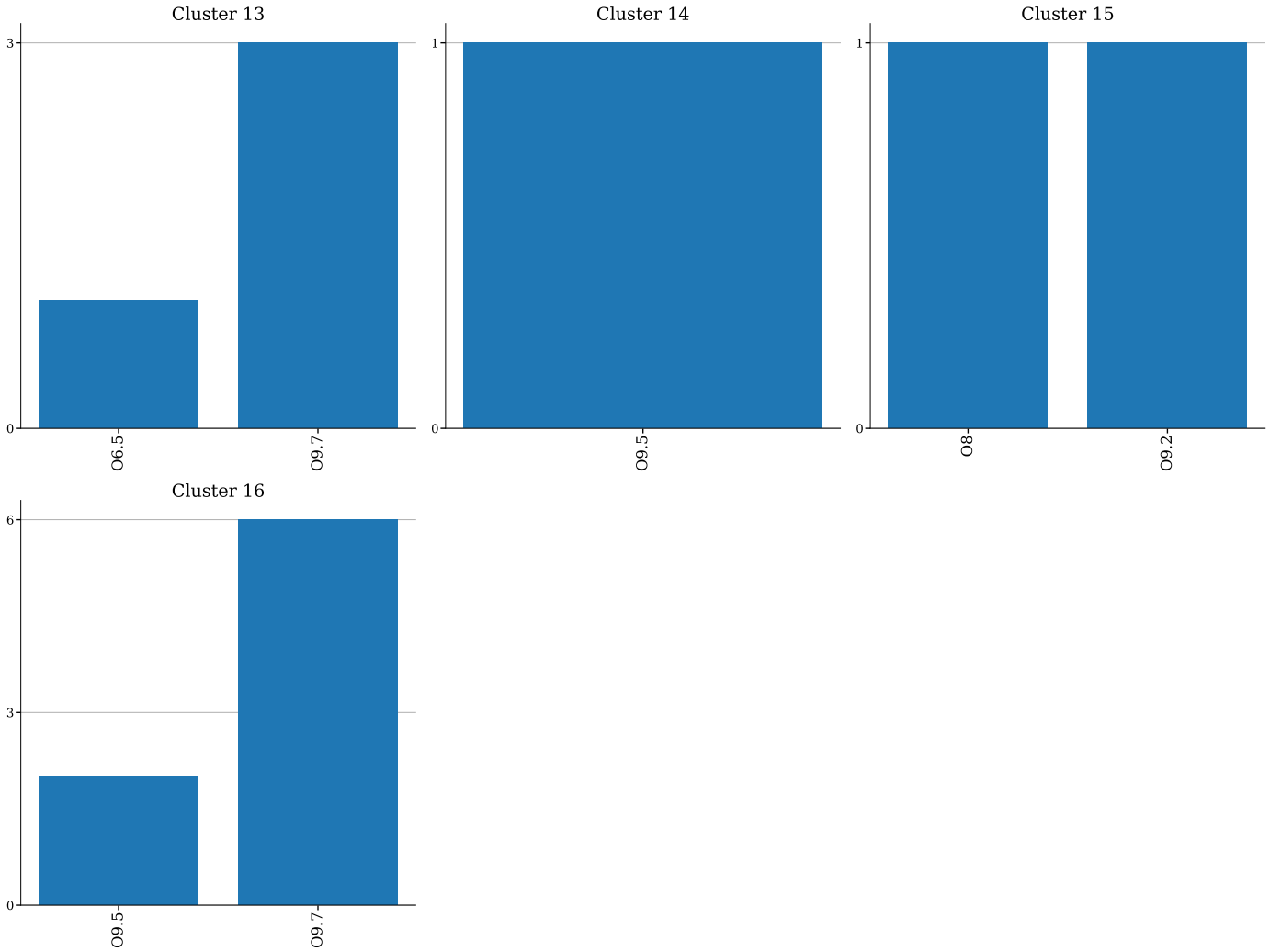


Figura 3.7: (continuação)

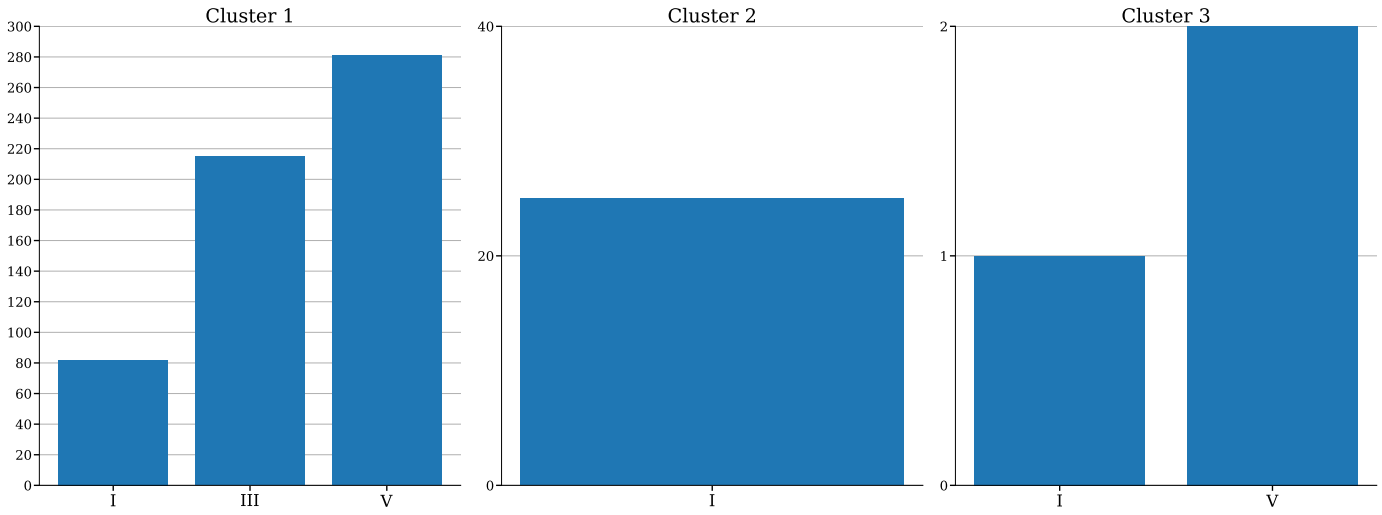


Figura 3.8: 606 estrelas, 3 *features* ($k = 3$), *score* $= 0.79$

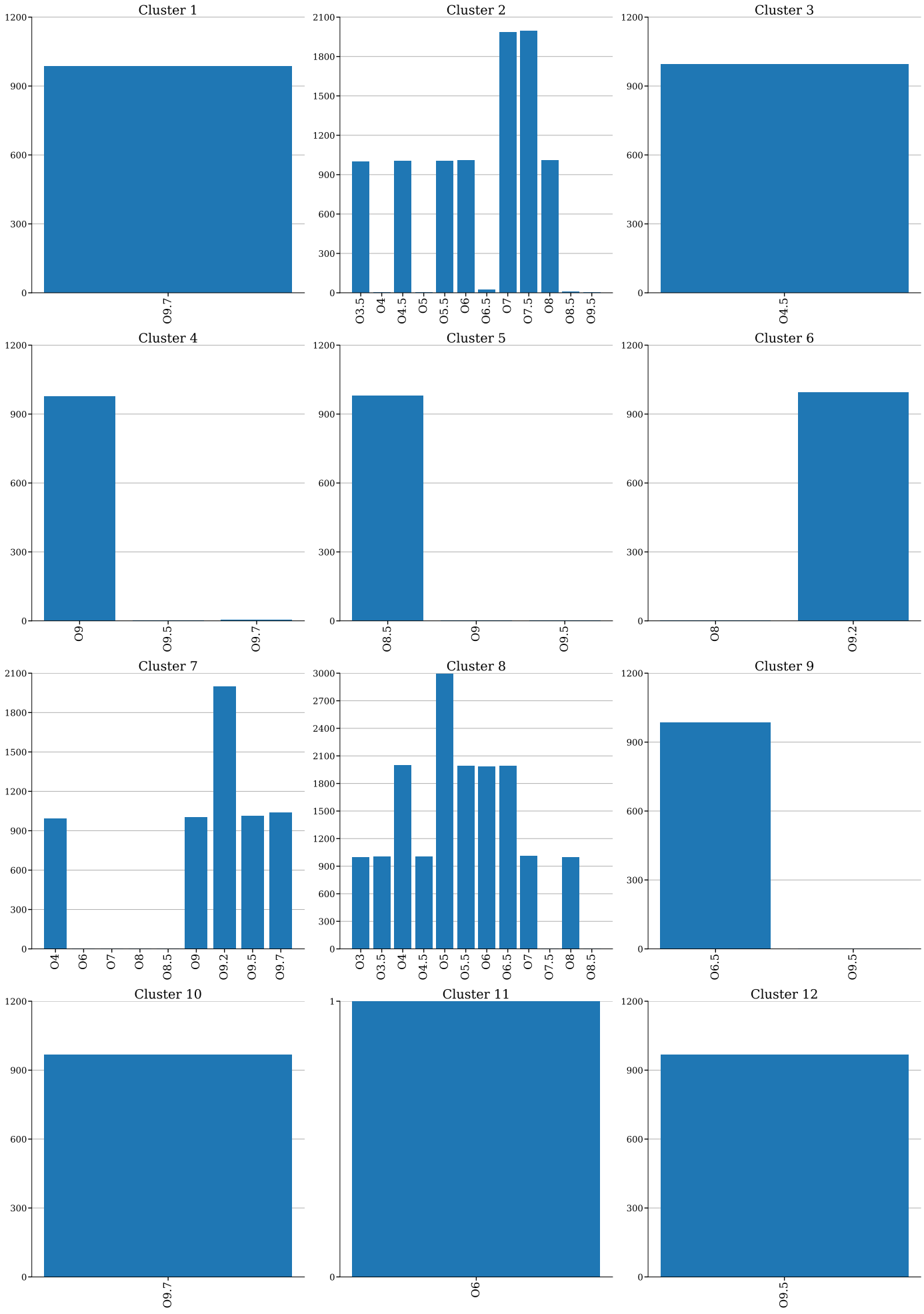


Figura 3.9: 47.000 estrelas, 4 *features* ($k = 16$), *score* $= 0.55$

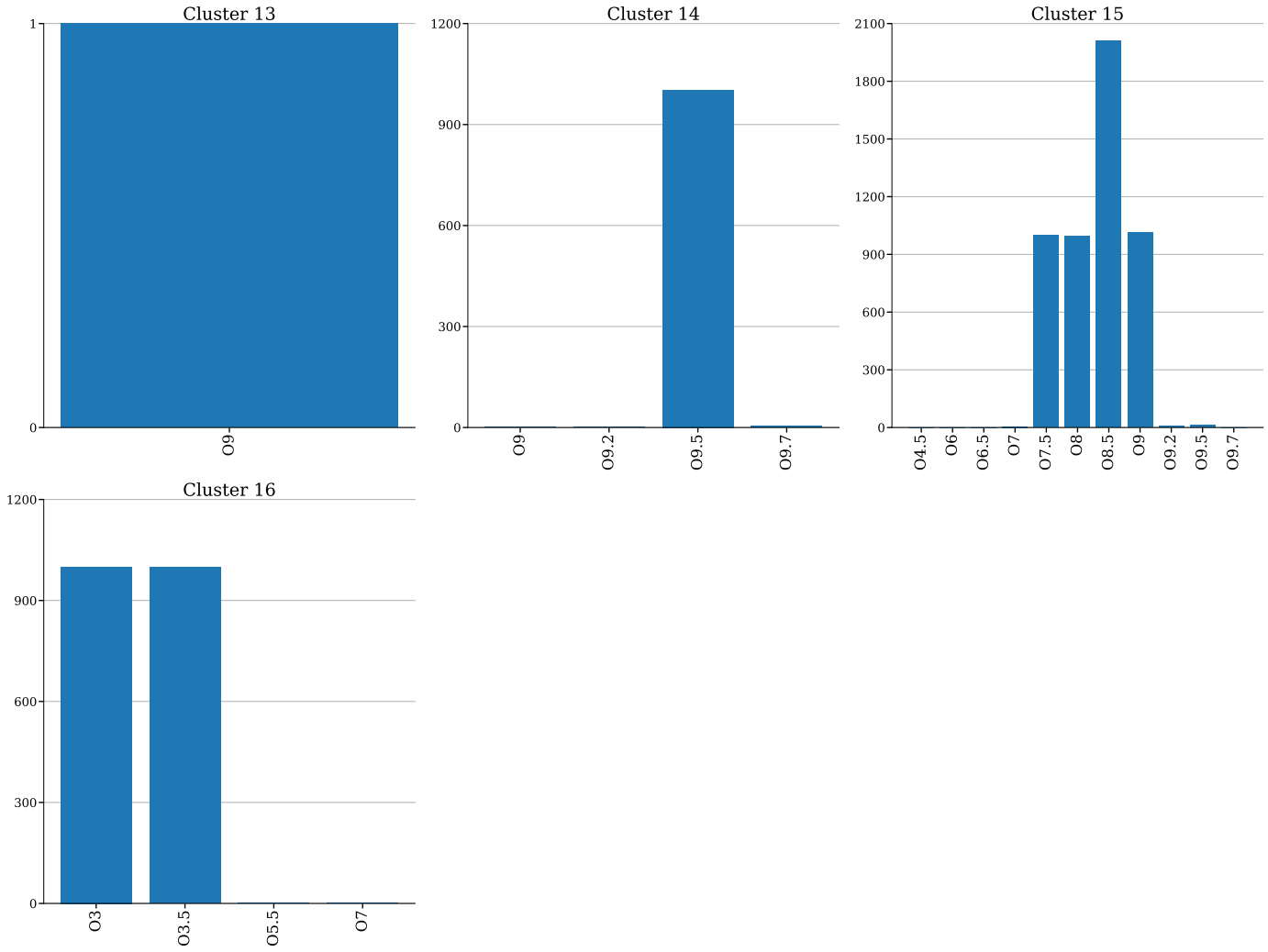


Figura 3.9: (continuação)

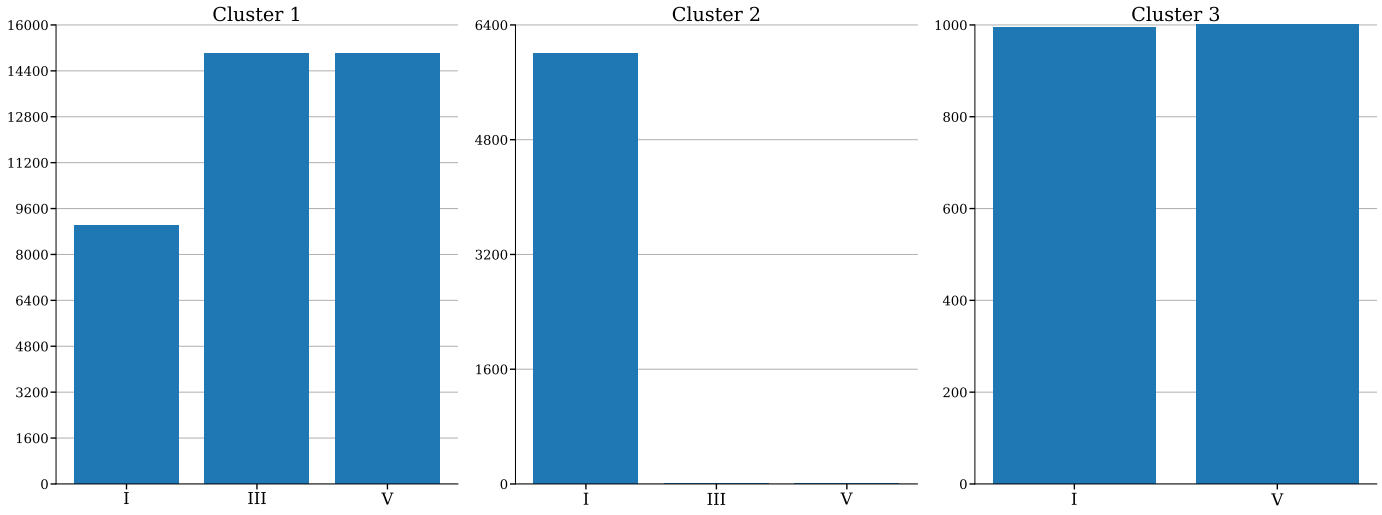
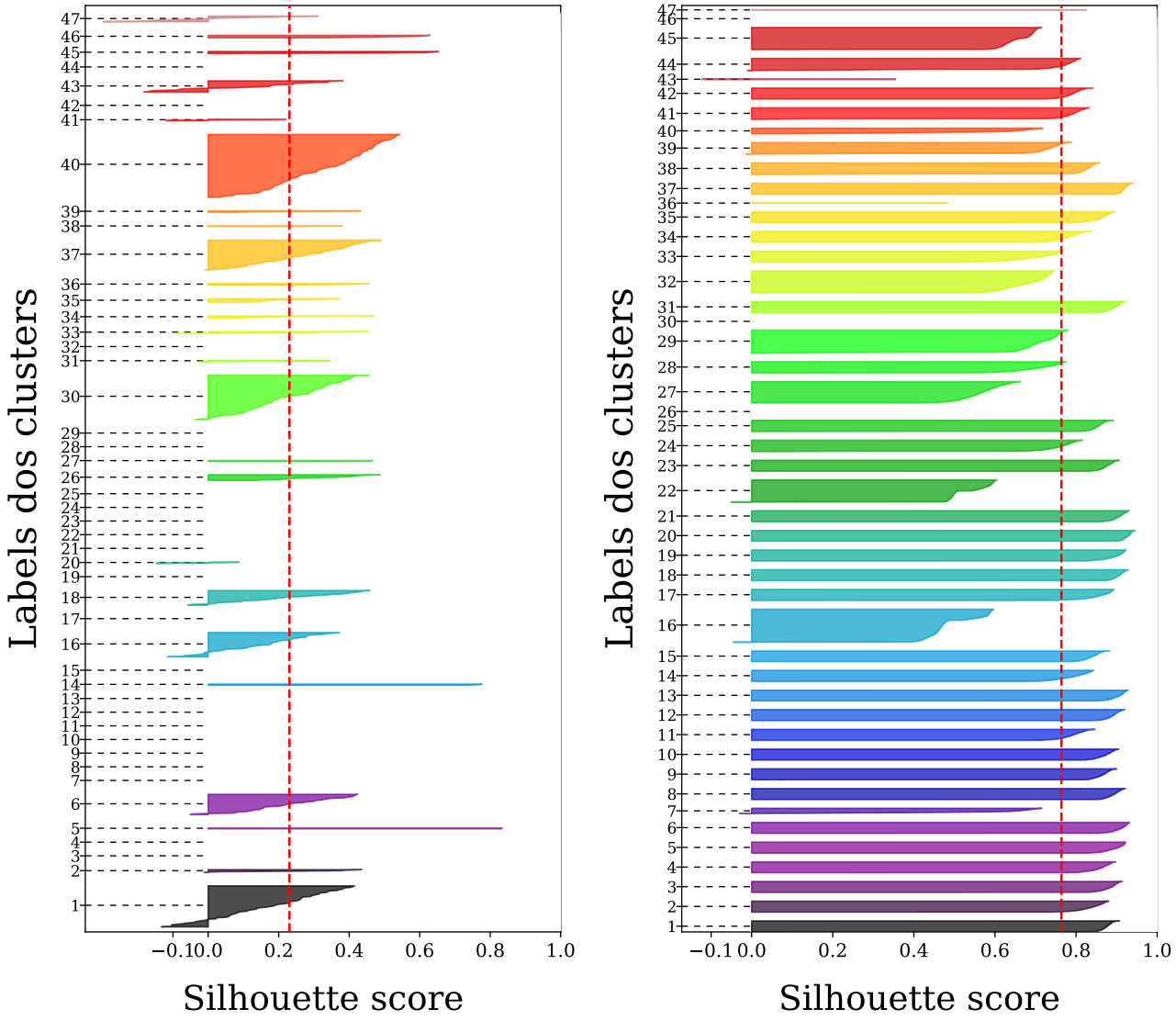


Figura 3.10: 47.000 estrelas, 3 *features* ($k = 3$), *score* $= 0.59$

Vimos que para o caso da classificação com base nas classes de luminosidade, o aumento do número de estrelas de 606 para 47.000 ocasionou na piora do *score*. Na Seção 3.2 iremos discutir sobre esta questão. Mas para reafirmar o ponto de que o aumento da quantidade de dados de 606 para 47.000 estrelas (amostra artificial) é algo positivo para nossa análise, apresentaremos uma visualização alternativa dos mesmos resultados das classificações apresentadas nas Figuras 3.5 e 3.6 através dos gráficos da Figura 3.11 a seguir. Neles, podemos comparar a qualidade da classificação dos elementos presentes em cada um dos *clusters* com o *score* médio para $k = 47$ (como mencionado na Subseção 2.3.1), ao mesmo tempo que temos uma visualização da qualidade da classificação como um todo por meio das barras coloridas. A Figura 3.11a evidencia uma série de problemas com a classificação realizada utilizando o *dataset* de 606 estrelas com 7 *features* e $k = 47$. Como o *score* médio é baixo, o fato de boa parte das estrelas terem alcançado um *score* individual acima da média não é um indicativo de boa classificação. Além do mais, fica claro que alguns poucos *clusters*, como os de *label* 1, 30 e 40, tem uma concentração maior de estrelas se comparados ao resto, especialmente aos *clusters* com apenas uma estrela, como 3, 9 e 42.

Já na Figura 3.11b, fica claro que a qualidade da classificação melhora drasticamente. A maioria dos *clusters* tem todas ou boa parte das estrelas com um *score* acima da média, que desta vez é alta. As estrelas também apresentam uma distribuição muito mais equilibrada entre os diferentes *clusters*, e nota-se que poucos deles contêm apenas uma estrela. A eliminação da sub-representação de certos tipos espectrais sem dúvida teve grande contribuição neste sentido. Entretanto, ainda existem estrelas com *score* negativo. Conforme visto anteriormente, pontos com *scores* negativos são um reflexo de casos em que a distância intra-*cluster* média é maior que a distância média até o segundo *cluster* mais próximo do ponto em questão, ou seja, o ponto é atribuído a um *cluster* mesmo estando mais próximo de outro. Dito isso, os *scores* negativos têm valores próximos de zero, o que provavelmente indica que, no contexto do espaço de *features*, estas estrelas estão situadas em regiões que apresentam sobreposição entre dois ou mais *clusters*, resultando numa classificação errônea, possivelmente porque o algoritmo não alcança uma convergência perfeita mesmo após várias iterações.



(a) 606 estrelas, 7 *features* ($k = 47$)

(b) 47.000 estrelas, 7 *features* ($k = 47$)

Figura 3.11: Cada uma das barras coloridas representa um *cluster*, onde as linhas pretas tracejadas indicam suas respectivas *labels*, dispostas ao longo do eixo das ordenadas. As barras faltantes indicam que o *cluster* que ocuparia aquela posição possui apenas uma estrela. Além disso, cada barra é composta por linhas horizontais que representam as estrelas do *cluster*, e o comprimento da linha está associado ao silhouette *score* individual da estrela, indicado ao longo do eixo das abscissas. Portanto, quanto maior a largura de uma barra colorida, mais estrelas fazem parte do *cluster*, e vice-versa. A linha tracejada vermelha indica o silhouette *score* médio para $k = 47$. Para fins de comparação, os *clusters* e *labels* dos gráficos (a) e (b) são os mesmos das Figuras 3.5 e 3.6, respectivamente.

Portanto, por meio das Figuras 3.5, 3.6, e 3.11 fica claro que aumentar a amostra de 606 para 47.000 estrelas (amostra artificial) quando usamos todas as 7 *features* e escolhemos $k = 47$, contribui substancialmente para a melhora da classificação com o k-means.

Na Seção 3.2 iremos analisar os problemas identificados em nossas classificações e discutir possíveis maneiras de solucioná-los.

3.2 Discussão de Problemas

Um dos problemas recorrentes nos casos de classificação envolvendo o *dataset* de 606 estrelas são *clusters* povoados por apenas uma ou poucas estrelas de certa classe. Embora uma maneira de mitigar esse problema seja o aumentar a quantidade de elementos das classes em questão, nem sempre haverá mais dados reais disponíveis e, além disso, aumentar artificialmente a quantidade de dados pode não ser algo apropriado, ou ser tarefa demasiadamente trabalhosa a depender da quantidade de *features* envolvidas. Idealmente, queremos que o algoritmo perceba que os pontos "solitários" pertencem a outros clusters. Identificar e remover tais pontos de nosso *dataset* pode não ser ideal, uma vez que estrelas de subtipos espectrais mais quentes não estão presentes em grande quantidade devido à sua raridade.

Uma possível explicação para a inesperada perda de desempenho das classificações envolvendo apenas as classes de luminosidade é o motivo por trás do critério de junção das classes II, III e IV, mencionado em 1.2.2 (ver Martins, 2018). Um dos intervalos de valores das larguras equivalentes da linha de He II λ4686 inclui estrelas II, III e IV, além de algumas de classe V. Outro intervalo inclui simultaneamente supergigantes e gigantes brilhantes. Este *overlap* pode "confundir" o algoritmo, uma vez que haverá sobreposição de estrelas de classes de luminosidade distintas no espaço de *features*. Portanto, é razoável supor que o aumento da quantidade de estrelas com estas características intensifica a sobreposição e exacerba o problema de identificação das classes, prejudicando o desempenho da classificação.

O problema mais fundamental e que sem dúvida afeta o desempenho das classificações em todos os casos discutidos neste trabalho é a dispersão e sobreposição dos pontos no espaço de *features*. Como nos casos expostos aqui utilizamos simultaneamente 3, 4 ou 7 *features*, a visualização dos pontos da amostra em gráficos não traria grandes benefícios, uma vez que só poderíamos gerá-los para visualizar as features relacionadas às classes de luminosidade, devido à tridimensionalidade do espaço de *features*. Encontrar um modo de identificar não só quais *features* se sobrepõem a quais, mas também os pontos que possam ser considerados *outliers*, traria a possibilidade de fazer uma análise mais específica de identificação e solucionamento de problemas que poderia posteriormente ser generalizada para todo o *dataset*.

Capítulo 4

Conclusão e Perspectivas

Neste trabalho, além de nos familiarizarmos com os critérios utilizados para classificação espectral de estrelas O, utilizamos espectros de 606 estrelas deste tipo extraídos do Galactic O-Star Catalogue e de diversos arquivos públicos (ELODIE, SOPHIE, CFHT Science, PolarBase e ESO). Medimos as larguras equivalentes das linhas espectrais relevantes e classificamos as estrelas utilizando o método k-means, um algoritmo de aprendizado de máquina não supervisionado que identifica *clusters* de classes distintas nos pontos distribuídos no espaço de *features*. Além disso, para testar efeitos do tamanho da amostra e possivelmente contornar o problema de sub-representatividade de alguns tipos de estrelas, geramos estrelas artificiais com base nas da amostra de Martins (2018), aumentando arbitrariamente o volume de dados de modo que cada uma das 47 classes contabilizasse 1.000 elementos, totalizando 47.000 estrelas (amostra artificial) e, em seguida, também as classificamos utilizando o k-means.

Verificamos que o algoritmo apresenta um bom potencial para classificação de espectros de estrela de alta massa. Demonstramos que uma amostra de dados grande e de boa qualidade é essencial. Por exemplo, se não soubéssemos a classificação prévia dos dados analisados, a separação de classes encontradas via algoritmo teria se mostrado extremamente útil no caso da amostra de 47.000 estrelas (amostra artificial), com k = 47 (sugerido via silhoutte score) e 7 *features*. Uma pré-classificação automática de milhares de espectros seria extremamente valiosa para uma análise de dados posterior, mais detalhada (e.g., com modelos de atmosferas),

de um *survey*. Por exemplo, encontramos neste trabalho *clusters* povoados praticamente com apenas uma classe, ou, em casos piores, *clusters* de classificações praticamente vizinhas em termos de tipo espectral, o que implica em propriedades físicas próximas. Além disso, a utilização de gráficos como os das Figuras 3.11a e 3.11b nos fornece uma visão geral do resultado em um único gráfico, como também ajuda na escolha de um valor para o hiperparâmetro k, complementando a análise de gráficos do silhouette *score* em função de k, bem como a análise por histogramas. Outro fator que pode ser levado em consideração é a possibilidade do algoritmo k-means encontrar padrões e relações inesperadas, o que poderia implicar em *clusters* povoados por estrelas até então sem uma conexão clara, mas que mereçam uma nova classificação na literatura, por exemplo.

Uma possível solução para o problema encontrado em que os *clusters* povoados com apenas uma ou poucas estrelas de certa classe (ver Figura 3.5) pode ser a redução do valor do hiperparâmetro k, que forçaria o algoritmo a "relocar" os pontos isolados para *clusters* mais povoados. Caso a redução do valor de k implique na diminuição da quantidade de *features*, a utilização de algoritmos de redução de dimensionalidade pode vir a ser útil.

Para melhorar ainda mais o desempenho do k-means nas classificações tanto com o *dataset* de 606 estrelas quanto com o de 47.000 para o caso em que $k = 47$, é interessante melhorar sua capacidade de identificação das classes de luminosidade. No entanto, não há outro critério na literatura que classifique melhor em termos de luminosidade e, portanto, juntar as classes II, III e IV ainda é o recomendado, apesar dos problemas que isto pode trazer para a classificação.

Fazer *plots* 2D e 3D de diferentes combinações de *features*, poderia ajudar a identificar problemas mencionados no capítulo anterior, como dispersão e sobreposição dos pontos. Alternativamente, utilizar um algoritmo de redução de dimensionalidade de modo que um *plot* com todas as *features* relevantes torne-se possível pode facilitar esta tarefa, além de contribuir para solucionar o problema dos *clusters* povoados por poucos elementos de certa classe.

Bibliografia

Auer, L., H. & Mihalas, D. (1972). *ApJS*, 24:193–246.

Barbá, R. H., Gamen, R., Arias, J. I., Morrell, N., Maíz Apellániz, J., Alfaro, E., Walborn, N., & Sota, A. (2010). *Revista Mexicana de Astronomía y Astrofísica Conference Series*, 38:30–32.

Bouchy, F., Díaz, R. F., Hébrard, G., Arnold, L., Boisse, I., Delfosse, X., Perruchot, S., & Santerne, A. (2013). *A&A*, 549, A49.

Bouchy, F., Hébrard, G., Udry, S., Delfosse, X., Boisse, I., Desort, M., Bonfils, X., Eggenberger, A., Ehrenreich, D., Forveille, T., Lagrange, A. M., Le Coroller, H., Lovis, C., Moutou, C., Pepe, F., Perrier, C., Pont, F., Queloz, D., Santos, N. C., Ségransan, & D. Vidal-Madjar, A. (2009). *A&A*, 505:853–858.

Clayton, D. D. (1983). *Principles of Stellar Evolution and Nucleosynthesis*. The University of Chicago Press.

Conti, P. S. & Alschuler, W. R. (1971). *ApJ*, 170:354–344.

Conti, P. S. & Frost, S. A. (1977). *ApJ*, 212:728–742.

Donati, J. F., Semel, M., Carter, B. D., Rees, D. E., & Collier Cameron, A. (1997). *MNRAS*, 291:658–682.

Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M. H., Brett, M., Haldane, A., del Río, J. F., Wiebe, M., Peterson, P., Gérard-Marchant, P.,

Sheppard, K., Reddy, T., Weckesser, W., Abbasi, H., Gohlke, C., & Oliphant, T. E. (2020). *Nature*, 585(7825):357–362.

Hunter, J. D. (2007). *Comput Sci Eng*, 9(3):90–95.

Jacoby, G. H., Hunter, D. A., & Christian, C. A. (1984). *ApJS*, 56:257–281.

Maravelias, G., Bonanos, A. Z., Tramper, F., de Wit, S., Yang, M., & Bonfini, P. (2022). *Accepted for publication in A&A*.

Martins, F. (2018). *A&A*, 616, A135.

Mathys, G. (1988). *A&AS*, 76:427–444.

Maíz Apellániz, J., Sota, A., Morrell, N. I., Barbá, R. H., Walborn, N. R., Alfaro, E. J., Gamen, R. C., Arias, J. I., & Gallego Calvente, A. T. (2013). *Massive Stars: From alpha to Omega, 198*.

Maíz Apellániz, J., Sota, A., Walborn, N. R., Alfaro, E. J., Barbá, R. H., Morrell, N. I., Gamen, R. C., & Arias, J. I. (2011). *Highlights of Spanish Astrophysics VI, ed. M. R. Zapatero Osorio, J. Gorgas, J. Maíz Apellániz, J. R. Pardo, & A. Gil de Paz, 467–472*.

McKinney, W. (2010). In *Proceedings of the 9th Python in Science Conference*, pages 56–61.

Morgan, W., W. & Keenan, P., C. (1973). *ARA&A*, 11:29–50.

Moultaka, J., Ilovaisky, S., Prugniel, P., & Soubiran, C. (2004a). *SF2A-2004: Semaine de l'Astrophysique Francaise, ed. F. Combes, D. Barret, T. Contini, F. Meynadier and L. Pagani, 547*.

Moultaka, J., Ilovaisky, S. A., Prugniel, P., & Soubiran, C. (2004b). *PASP*, 116:693–698.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). *JMLR*, 12:2825–2830.

Petit, P., Louge, T., Théado, S., Paletou, F., Manset, N., Morin, J., Marsden, S. C., & Jeffers, S. V. (2014). *PASP*, 126:469.

Reback, J., McKinney, W., jbrockmendel, Van den Bossche, J., Augspurger, T., Cloud, P., gfyoung, Sinhrks, Hawkins, S., Klein, A., Roeschke, M., Tratner, J., Petersen, T., She, C., Ayd, W., MomIsBestFriend, Garcia, M., Schendel, J., Hayden, A., Saxton, D., Jancauskas, V., McMaster, A., Battiston, P., Seabold, S., chris-b1, h-vetinari, Dong, K., Hoyer, S., Overmeire, W., & Winkel, M. (2020). pandas-dev/pandas: Pandas.

Seeds, M. A. & Backman, D. E. (2011). *Foundations of Astronomy.* Brooks/Cole, Cengage Learning.

Sota, A., Maíz Apellániz, J., Walborn, N. R., Alfaro, E. J., Barbá, R. H., Morrell, N. I., Gamen, R. C., & Arias, J. I. (2011). *ApJS*, 193, 24.

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., Carey, C. J., Polat, İ., Feng, Y., Moore, E. W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E. A., Harris, C. R., Archibald, A. M., Ribeiro, A. H., Pedregosa, F., van Mulbregt, P., & SciPy 1.0 Contributors (2020). *Nat Methods*, 17:261–272.

Apêndice A

Tabela dos dados utilizados

O propósito deste apêndice é mostrar algumas das estrelas utilizadas neste trabalho com suas respectivas *features* e tipos espectrais, de modo a deixar mais claro a maneira como estruturamos nossos dados. A Tabela A.1 demonstra 5 das 606 estrelas observadas, enquanto a Tabela A.2 demonstra 5 das 46.394 estrelas artificiais. Vale ressaltar que algumas colunas não foram incluídas por questões de clareza e de tamanho, e que embora as estrelas artificiais estejam em uma tabela separada, ao realizar as classificações com 47.000 estrelas os dados de ambas as tabelas são utilizados.

Tabela A.1: Amostragem dos Dados Observados

Nº	ID	$\frac{W_\lambda[\text{He I } \lambda4471]}{W_\lambda[\text{He II } \lambda4542]}$	$\frac{W_\lambda[\text{He I } \lambda4144]}{W_\lambda[\text{He II } \lambda4200]}$	$\frac{W_\lambda[\text{He I } \lambda4388]}{W_\lambda[\text{He II } \lambda4542]}$	$\frac{W_\lambda[\text{Si III } \lambda4552]}{W_\lambda[\text{He II } \lambda4542]}$	$\frac{W_\lambda[\text{He II } \lambda4686]}{W_\lambda[\text{He I } \lambda4713]}$	$\frac{W_\lambda[\text{Si IV } \lambda4089]}{W_\lambda[\text{He I } \lambda4026]}$	$W_\lambda[\text{He II } \lambda4686]$	Tipo Espectral
1	1 Cam A	4.290388	1.151525	1.973739	0.651126	0.675952	0.646722	0.392651	O9.7III
2	29 CMa	1.252283	-0.094385	0.314964	0.018694	14.502652	0.299880	-2.227684	O7I
...	...	...	...	...	...	...	...	...	...
604	θ Mus B	1.859876	0.849338	1.062964	0.320518	3.496933	0.498831	0.617889	O9III
605	ζ Ori AaAb	2.430143	1.041997	1.512673	0.424104	0.644539	0.849426	0.242554	O9.2I
606	ζ Ori B	3.246030	1.197645	2.180169	0.780349	1.137502	0.618838	0.245962	O9.7III

Tabela A.2: Amostragem dos Dados Artificiais

Nº	ID	$\frac{W_\lambda[\text{He I } \lambda4471]}{W_\lambda[\text{He II } \lambda4542]}$	$\frac{W_\lambda[\text{He I } \lambda4144]}{W_\lambda[\text{He II } \lambda4200]}$	$\frac{W_\lambda[\text{He I } \lambda4388]}{W_\lambda[\text{He II } \lambda4542]}$	$\frac{W_\lambda[\text{Si III } \lambda4552]}{W_\lambda[\text{He II } \lambda4542]}$	$\frac{W_\lambda[\text{He II } \lambda4686]}{W_\lambda[\text{He I } \lambda4713]}$	$\frac{W_\lambda[\text{Si IV } \lambda4089]}{W_\lambda[\text{He I } \lambda4026]}$	$W_\lambda[\text{He II } \lambda4686]$	Tipo Espectral
1	—	5.862170	1.376965	4.189531	0.892068	5.862256	2.627744	1.190069	O9.7III
2	—	6.745441	1.403918	3.559488	0.855437	6.355786	2.870501	1.331899	O9.7III
...	...	...	...	...	...	...	...	...	...
46392	—	0.535857	0.089600	0.087265	0.208257	7.610537	-0.195330	0.551327	O3.5V
46393	—	0.595216	0.080836	0.075400	0.200354	7.388161	-0.183303	0.533894	O3.5V
46394	—	0.598465	0.083413	0.083313	0.181763	7.208217	-0.210336	0.559066	O3.5V

Apêndice B

O algoritmo k-means

Do modo como está abaixo, o código faz a classificação de 47.000 estrelas utilizando todas as 7 *features* ($k = 47$). Para os outros casos, apenas adaptamos o código nas linhas 6 e 13. O *dataset* utilizado possui colunas para as *features*, larguras equivalentes, tipos espectrais, classes de luminosidade, e identificadores de catálogo (GOSC ou Martins). No Apêndice A, algumas destas colunas podem ser visuzalidas em duas tabelas reduzidas.

```
1  from scipy.cluster.vq import whiten
2  from sklearn.preprocessing import StandardScaler
3  from sklearn.cluster import KMeans
4
5  #A linha abaixo cria um array contendo os valores das features de todas as estrelas no
       dataset
6  x = np.array(dataset47000[['4471/4542','4144/4200','4388/4542','4552/4542','4686','4686/4713
       ','4089/4026']])
7
8  #Abaixo, duas operações normalizadoras são feitas: "whiten" as realiza entre features
       distintas de uma única estrela e "StandardScaler" entre todas as estrelas mas numa
       feature por vez
9  x = whiten(x)
10 scaler = StandardScaler()
11 x = scaler.fit_transform(x)
```

```
12 #A linha 14 prepara o algoritmo k-means. "n_clusters" define o valor de k. "n_init" define a
       quantidade de iterações com centroides iniciais distintos, onde o resultado mais bem-
       sucedido é selecionado. "max_iter" define a quantidade máxima de vezes que as etapas de
       convergência se repetirão para uma única iteração. "init" define o critério de escolha
       das posições iniciais dos centroides
13 kmeans = KMeans(n_clusters = 47, n_init=100, max_iter=300, init='k-means++')
14 kmeans.fit(x) #aplica o k-means aos dados
```

Após rodar o código acima, o resultado já estará disponível para ser analisado do modo que for mais conveniente. Adicionalmente, os atributos `labels_` e `cluster_centers_`, por exemplo, podem ser usados para retornar *arrays* com as *labels* dos *clusters* encontrados para cada estrela ou com as posições finais dos centroides de cada *cluster*, respectivamente. Mais detalhes e funcionalidades estão disponíveis na página da função KMeans.