Trabalho Prático de Mineração de Dados 2021-1

Camila A. Paiva[1] e Fábio G. Santos[2]

[1,2] Instituto de Computação – Universidade Federal Fluminense (UFF)
Niterói – RJ – Brazil

{fabiogomes,camilapaiva}@id.uff.br

Abstract. *This work aims to show the results gathered from the practical work of data mining class. It requires execution of three tasks of data mining on a database file previously chosen. These tasks are: classification, associate rules extraction and clustering. For each task, different approaches and input parameters are required.*

Resumo. *Este trabalho tem como objetivo mostrar os resultados obtidos a partir do trabalho prático da aula de mineração de dados. Requer a execução de três tarefas de mineração de dados em um arquivo de banco de dados previamente escolhido. Essas tarefas são: classificação, extração de regras de associação e clustering. Para cada tarefa, diferentes abordagens e parâmetros de entrada são necessários.*

1. Introdução

Por milhares de anos, os seres humanos tentam entender como o cérebro é capaz de perceber, entender, prever e manipular um mundo muito maior e mais complicado do que ele. O campo da inteligência artificial, ou IA, vai ainda mais longe: a inteligência artificial visa entender e construir entidades inteligentes [Russell et al. 2010].

A IA é um dos mais novos campos da ciência e engenharia. Russell et al. afirmam que seu desenvolvimento iniciou logo após a Segunda Guerra Mundial e foi nomeada por volta de 1956. Atualmente, tal campo abrange uma grande variedade de subcampos, tais como, provar teoremas matemáticos, diagnosticar doenças, dirigir carros e jogar xadrez. A IA é relevante para qualquer tarefa intelectual, é verdadeiramente um campo universal.

Existem diversas definições para a IA, dentre elas: o estudo dos cálculos que permitem perceber, argumentar e agir [Winston 1992] ou o estudo das faculdades mentais através do uso de modelos computacionais [Winston 1992].

Nos últimos anos a IA ganhou cada vez mais popularidade. Com a globalização e a informatização dos processos, as organizações geram um grande número de dados diariamente, também chamado de *Big Data* [Simon 2015]. Através de campos, como a IA, algoritmos podem ser utilizados para realizar previsões ou sugestões calculadas com base em um conjunto de dados. Alguns dos exemplos mais comuns de aprendizagem de máquina são: os algoritmos utilizados na plataforma da Netflix, que sugerem filmes baseados em escolhas passadas, ou algoritmos da Amazon que recomendam livros com base em livros comprados anteriormente.

Segundo Russell et al., os algoritmos de aprendizado de máquina podem ser divididos em 3 categorias: aprendizagem supervisionada, aprendizagem não

supervisionada e aprendizagem por reforço. Na aprendizagem supervisionada, são apresentados exemplos de entradas e saídas desejadas (pares *input-output*). Dessa forma, tais algoritmos aprendem uma função, ou regra geral, que prevê a saída para novas entradas. Para a categoria aprendizado não supervisionado, nenhum tipo de saída é dado aos algoritmos, deixando-o sozinho encontrar um padrão nas entradas fornecidas. Por fim, na aprendizagem tem como foco a criação de agentes capazes de tomar decisões acertadas em um ambiente sem que se tenha qualquer conhecimento prévio sobre o tal ambiente.

Os Algoritmos de Aprendizagem de Máquina são aplicados em diversas áreas e para diferentes fins, um exemplo dessa aplicação pode ser constatado em [Bramer 2016], o qual aplica a técnica *S-Transform (ST)* para extrair recursos de sons cardíacos. Esses recursos são utilizados como entrada para o classificador, *Multilayer Perceptron Network*. Como resultados, os autores obtiveram 98% de acerto na classificação. Neste mesmo sentido, este trabalho propõe o uso de três abordagens de mineração de dados sobre um conjunto de informações obtidas através do repositório de *Machine Learning do UCI* [Dua, D. and Graff, C. 2019]. A aplicação destas abordagens tem como objetivo comparar diferentes técnicas e expor seus respectivos resultados. Dentre as abordagens estão: classificação, extração de regras de associação e clusterização.

2. Base De Dados

O conjunto de dados utilizado trata-se de uma base sobre vinhos[1]. Tais informações são resultantes de uma análise química sobre os vinhos oriundos de uma mesma região na Itália, mas de diferentes espécies. Tal análise determinou 13 constituintes. A base é composta por 178 instâncias distribuídas entre três classes (classe 1 com 59 instâncias, classe 2 com 71 instâncias e classe 3 com 48 instâncias) e 13 atributos numéricos que são: *Alcohol, Malic acid, Ash, Alcalinity of ash, Magnesium, Total phenols, Flavanoids, Nonflavanoid phenols, Proanthocyanins, Color intensity, Hue, OD280/OD315 of diluted wines, Proline.*

A Tabela 1 é composta por todos os atributos numéricos da base escolhida com seus respectivos domínios, médias dos valores e desvio padrão. Percebe-se que, atributos como: *Nonflavanoid phenols, Hue e Proanthocyanins* possuem o desvio padrão baixo, ou seja, tais atributos apresentam dispersões mais uniformes que os demais.

Tabela 1. Resumo da validação - Random Forest número de árvores em 10

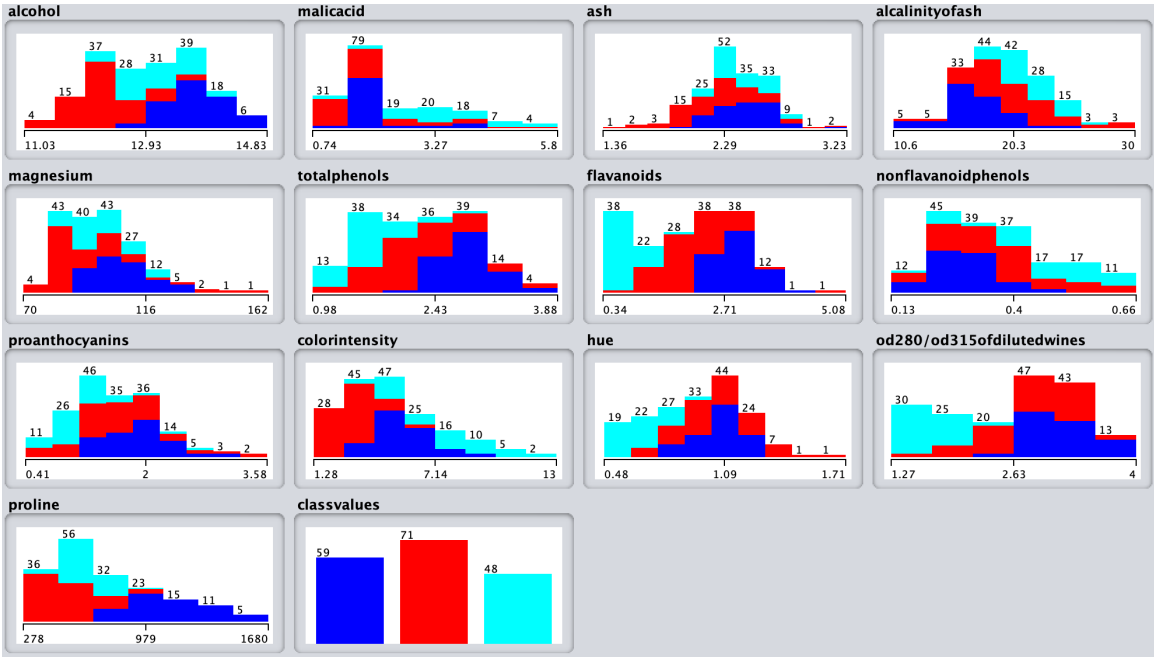
Atributo	Domínio	Média	Desvio Padrão
Alcohol	[11.03 - 14.83]	13.0	0.8
Malic acid	[0.74 - 5.8]	2.34	1.12
Ash	[1.36 - 3.23]	2.36	0.27
Alcalinity of ash	[10.6 - 30]	19.5	3.3
Magnesium	[70 - 162]	99.7	14.3

[1] https://archive.ics.uci.edu/ml/datasets/Wine

Total phenols	[0.98 - 3.88]	2.29	0.63
Flavanoids	[0.34 - 5.08]	2.03	1.00
Nonflavanoid phenols	[0.13 - 0.66]	0.36	0.12
Proanthocyanins	[0.41 - 3.58]	1.59	0.57
Color intensity	[1.28 - 13]	5.1	2.3
Hue	[0.48 - 1.71]	0.96	0.23
OD280/OD315 of diluted wines	[1.27 - 4]	2.61	0.71
Proline	[278 - 1680]	746	315

A Figura 1 mostra a distribuição dos atributos em relação às classes 1, 2 e 3 da base de dados através de histogramas gerados pela ferramenta Weka[2]. Não foi necessária a realização de um pré-processamento, tais como, limpeza dos dados, retirada de atributos com possíveis ruídos e atributos que poderiam interferir nos resultados como, por exemplo, IDs.

Figura 1. Distribuição dos atributos



3. Classificação

A classificação é utilizada para classificar itens de um conjunto em classe ou grupos predefinidos. A tarefa de classificação requer a análise dos dados com a qual um modelo é construído para prever novos itens. Tal função atribui itens à uma coleção de

[2] https://www.cs.waikato.ac.nz/ml/weka/

categorias ou classes. O objetivo é prever com precisão a classe de destino para cada novo caso [Kesavaraj and Sukumaran 2013].

Três técnicas de classificação foram utilizadas neste trabalho, são elas: *RandomForest, Naive Bayes e KNN*. Para cada uma delas, os parâmetros de entrada foram ajustados para verificar se ocorreu ganho/aumento nos resultados da classificação.

3.1. RandomForest

RandomForest é uma técnica de classificação que utiliza um conjunto de N árvores com intuito de ter uma melhor acurácia final. Cada árvore desse conjunto executa a classificação de forma independente e a classe que for escolhida pela maioria das árvores é a classe resultante do processo de classificação.

Para a técnica de *RandomForest*, ajustou-se dois parâmetros com o objetivo de verificar a forma que eles interferem no modelo gerado, tais parâmetros são: número de árvores e subconjunto aleatório de atributos. Todos os testes foram realizados com 178 registros da base e técnica de validação *k-fold cross-validation* com valor igual a 10 e *full training set*.

3.1.1. Ajustando o número de árvores

O objetivo desse ajuste é verificar como o modelo melhora ou piora de acordo com o valor usado para o número de árvores no processamento. Pode-se verificar uma pequena alteração na acurácia quando aumentou-se de 10 para 100 o valor do parâmetro e para valores maiores que 100 não foi observado ganho na acurácia.

1. **Valor do parâmetro em 10**

Tabela 2. Resumo da validação - Random Forest número de árvores em 10

Correctly Classified Instances	174 (97.7528%)
Incorrectly Classified Instances	4 (2.2472 %)
Kappa statistic	0.966
Mean absolute error	0.0648
Root mean squared error	0.1447
Relative absolute error	14.7579 %
Root relative squared error	30.8829 %

Tabela 3. Detalhamento de acurácia - Random Forest número de árvores em 10

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	1,000	0,017	0,967	1,000	0,983	0,975	0,999	0,997	1
	0,944	0,000	1,000	0,944	0,971	0,954	0,996	0,993	2
	1,000	0,015	0,960	1,000	0,980	0,972	0,998	0,994	3
Média ponderada	0,978	0,010	0,978	0,978	0,977	0,966	0,998	0,995	

Tabela 4. Matriz de Confusão - Random Forest número de árvores em 10

a	b	c	Classificado como
59	0	0	a=1
2	67	2	b=2
0	0	48	c=3

2. Valor do parâmetro em 100

Tabela 5. Resumo da validação - Random Forest número de árvores em 100

Correctly Classified Instances	175 (98.3146 %)
Incorrectly Classified Instances	3 (1.6854 %)
Kappa statistic	0.9745
Mean absolute error	0.0655
Root mean squared error	0.1304
Relative absolute error	14.92 %
Root relative squared error	27.8283 %

Tabela 6. Detalhamento de acurácia - Random Forest número de árvores em 100

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	1,000	0,008	0,983	1,000	0,992	0,987	1,000	0,999	1
	0,958	0,000	1,000	0,958	0,978	0,965	0,999	0,998	2
	1,000	0,015	0,960	1,000	0,980	0,972	1,000	0,999	3
Média ponderada	0,983	0,007	0,984	0,983	0,983	0,974	0,999	0,999	

Tabela 7. Matriz de Confusão - Random Forest número de árvores em 100

a	b	c	Classificado como
59	0	0	a=1
1	68	2	b=2
0	0	48	c=3

3.1.2. Ajustando o número de subconjunto aleatório de atributos

Para um segundo teste, ajustou-se o número de subconjuntos aleatórios de atributos para verificar como o modelo melhora ou piora. Ao aumentar esse valor, verificou-se uma diminuição na acurácia de 174 para 173 mostrando que o aumento do valor para esse parâmetro não contribui para obter um melhor resultado de classificação.

1. Valor do parâmetro em 5

Tabela 8. Resumo da validação - Random Forest número de atributos aleatórios em 5

Correctly Classified Instances	174 (97.7528%)
Incorrectly Classified Instances	4 (2.2472 %)
Kappa statistic	0.9659
Mean absolute error	0.0788
Root mean squared error	0.143
Relative absolute error	17.9483%
Root relative squared error	30.5278 %

Tabela 9. Detalhamento de acurácia - Random Forest número de atributos aleatórios em 5

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0,983	0,008	0,983	0,983	0,983	0,975	0,999	0,999	1
	0,958	0,009	0,986	0,958	0,971	0,953	0,998	0,997	2
	1,000	0,015	0,960	1,000	0,980	0,972	1,000	0,999	3
Média ponderada	0,978	0,011	0,978	0,978	0,977	0,965	0,999	0,999	

Tabela 10. Matriz de Confusão - Random Forest número de atributos aleatórios em 5

a	b	c	Classificado como
58	1	0	a=1
1	68	2	b=2
0	0	48	c=3

2. Valor do parâmetro em 10

Tabela 11. Resumo da validação - Random Forest número de atributos aleatórios em 10

Correctly Classified Instances	173 (97.191%)
Incorrectly Classified Instances	5 (2.809%)
Kappa statistic	0.9575
Mean absolute error	0.0799
Root mean squared error	0.1524
Relative absolute error	18.2042%
Root relative squared error	32.5219%

Tabela 12. Detalhamento de acurácia - Random Forest número de atributos aleatórios em 10

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0,983	0,008	0,983	0,983	0,983	0,975	0,999	0,998	1
	0,944	0,009	0,985	0,944	0,964	0,942	0,996	0,994	2
	1,000	0,023	0,941	1,000	0,970	0,959	0,999	0,997	3
Média ponderada	0,972	0,013	0,973	0,972	0,972	0,957	0,998	0,996	

Tabela 13. Matriz de Confusão - Random Forest número de atributos aleatórios em 10

a	b	c	Classificado como
58	1	0	a=1
1	67	3	b=2
0	0	48	c=3

3.2. Naive Bayes

Segundo [Russell et al. 2010] o modelo de Rede Bayesiana mais comum usado na aprendizagem de máquinas é o modelo *Naive Bayes*. Esse modelo tem sido amplamente estudado desde os anos de 1950. Pode ser descrito como: a variável C (que deve ser predita) é a raiz e as variáveis X, atributos, são as folhas. O modelo é "*Naive*", pois assume que os atributos X são condicionalmente independentes uns dos outros, dada a classe C. Assumindo o conjunto de atributos , sendo o número de atributos, os parâmetros são calculados através da equação representada na Figura 2, Onde k são as possibilidades de resultados possíveis ou variáveis .

Figura 2. Probabilidade Ck tal que x1, …, xn

$$p(C_k|x_1, \dots, x_n)$$

Uma vez que o modelo foi treinado, pode ser usado para classificar novos exemplos para os quais a variável C não é observada. Assumindo o conjunto de atributos, sendo o número de atributos, a probabilidade de cada classe é dada pela equação representada na Figura 3.

Figura 3. Equação para cálculo da probabilidade de uma classe

$$P(C|x_1, \dots, x_n) = \alpha P(C) \prod_i P(x_i|C)$$

Para [Russell et al. 2010] o modelo *Naive Bayes* revela-se surpreendentemente bom em uma ampla gama de aplicações, além disso, possui alta escalabilidade em problemas muito grandes. Por fim, tal modelo de aprendizagem não tem dificuldade com dados ruidosos ou ausentes e podem fornecer previsões probabilísticas quando apropriado.

Para a técnica de *Naive Bayes* ajustou-se apenas um parâmetro para verificar de que forma ele interfere no modelo gerado: discretização supervisionada. Todos os testes contam com 178 registros na base e técnica de validação *k-fold cross-validation* com valor igual a 10 e *full training set*.

Com o ajuste do parâmetro de discretização supervisionada obteve-se um aumento na acurácia de aproximadamente 2%.

1. Valor do parâmetro em false

Tabela 14. Resumo da validação

Correctly Classified Instances	172 (96.6292 %)
Incorrectly Classified Instances	6 (3.3708 %)
Kappa statistic	0.9489
Mean absolute error	0.0217
Root mean squared error	0.1294
Relative absolute error	4.9371%
Root relative squared error	27.6176 %

Tabela 15. Detalhamento de acurácia

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0,949	0,000	1,000	0,949	0,974	0,962	0,998	0,997	1
	0,958	0,028	0,958	0,958	0,958	0,930	0,997	0,995	2
	1,000	0,023	0,941	1,000	0,970	0,959	1,000	1,000	3
Média ponderada	0,966	0,017	0,967	0,966	0,966	0,948	0,998	0,997	

Tabela 16. Matriz de Confusão

a	b	c	Classificado como
56	3	0	a=1
0	68	3	b=2
0	0	48	c=3

2. Valor do parâmetro em true

Tabela 17. Resumo da validação

Correctly Classified Instances	176 (98.8764 %)
Incorrectly Classified Instances	2 (1.1236 %)
Kappa statistic	0.983
Mean absolute error	0.0146
Root mean squared error	0.092
Relative absolute error	3.3229%
Root relative squared error	19.6443 %

Tabela 18. Detalhamento de acurácia

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	1,000	0,008	0,983	1,000	0,992	0,987	0,999	0,998	1
	0,972	0,000	1,000	0,972	0,986	0,977	0,998	0,997	2
	1,000	0,008	0,980	1,000	0,990	0,986	1,000	0,999	3
Média ponderada	0,989	0,005	0,989	0,989	0,989	0,983	0,999	0,998	

Tabela 19. Matriz de Confusão

a	b	c	Classificado como
59	3	0	a=1
1	69	1	b=2
0	0	48	c=3

3.3. K-Nearest Neighbours

O KNN é uma técnica de classificação *lazy* e, por esse motivo, se difere das anteriores utilizadas. Para cada nova instância, o algoritmo executa todo o processo de classificação. Enquanto que no *Naive Bayes* e *RandomForest*, o modelo é criado em um primeiro momento e depois as novas instâncias são classificadas utilizando esse modelo. O KNN utiliza a ideia de vizinhança, ou seja, verifica os K vizinhos mais próximos baseando-se no cálculo de distância para classificar uma nova instância. A distância pode ser calculada utilizando algumas abordagens diferentes, inclusive, a distância Euclidiana, calculada por: $dist(X1, X2) = \sqrt{\sum_{i=1}^{n} (X1_i - X2_i)^2}$.

Para a técnica de KNN, foram realizados alguns ajustes no parâmetro K para verificar de que forma ele interfere nos resultados gerados. Inicialmente, o valor atribuído foi K = 1 e testes foram realizados a partir deste. O melhor resultado

encontrado foi para k = 20. A partir de 20, a acurácia começou a decair. Todos os testes contaram com 178 registros e a técnica de validação *k-fold cross-validation* com valor igual a 10 e *full training set*.

O ajuste no parâmetro K fez com que o número de instâncias corretamente classificadas aumentasse de 169 para 174, melhorando assim a performance do classificador

1. **Valor do parâmetro em 1**

Tabela 20. Resumo da validação

Correctly Classified Instances	169 (94.9438 %)
Incorrectly Classified Instances	9 (5.0562 %)
Kappa statistic	0.9238
Mean absolute error	0.0413
Root mean squared error	0.1821
Relative absolute error	9.3973 %
Root relative squared error	38.8682 %

Tabela 21. Detalhamento de acurácia

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	1,000	0,042	0,922	1,000	0,959	0,940	0,983	0,936	1
	0,873	0,000	1,000	0,873	0,932	0,897	0,941	0,929	2
	1,000	0,031	0,923	1,000	0,960	0,946	0,983	0,917	3
Média ponderada	0,949	0,022	0,953	0,949	0,949	0,925	0,966	0,928	

Tabela 22. Matriz de Confusão

a	b	c	Classificado como
59	0	0	a=1
5	62	4	b=2
0	0	48	c=3

2. **Valor do parâmetro em 20**

Tabela 23. Resumo da validação

Correctly Classified Instances	174 (97.7528 %)
Incorrectly Classified Instances	4 (2.2472 %)
Kappa statistic	0.966
Mean absolute error	0.0629

Root mean squared error	0.148
Relative absolute error	14.3274 %
Root relative squared error	31.5855 %

Tabela 24. Detalhamento de acurácia

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	1,000	0,017	0,967	1,000	0,983	0,975	0,999	0,996	1
	0,944	0,000	1,000	0,944	0,971	0,954	0,997	0,994	2
	1,000	0,015	0,960	1,000	0,980	0,972	1,000	0,999	3
Média ponderada	0,978	0,010	0,978	0,978	0,977	0,966	0,998	0,996	

Tabela 25. Matriz de Confusão

a	b	c	Classificado como
59	0	0	a=1
2	67	2	b=2
0	0	48	c=3

3.4. Comparativo das técnicas de classificação

Após a realização das execuções de três técnicas de classificação sobre a base de dados com alguns ajustes nos parâmetros de entrada, pode-se observar que, de uma forma geral, foram obtidos bons resultados no contexto da base utilizada. Como mostra a Tabela 26, os valores para a acurácia e a *F-Measure* dos classificadores foram altos. A Acurácia teve variações entre 94,94 e 98,87 e a F-Measure entre 0,949 e 0,983.

Tabela 26. Comparativo de acurácia

Classificador	Accuracy	Precision	Recall	F-Measure
RandomForest I (parâmetro = 10)	97,75	0,978	0,978	0,977
RandomForest II (parâmetro = 100)	98,31	0,984	0,983	0,983
RandomForest III (parâmetro = 5)	97,75	0,978	0,978	0,977
RandomForest IV (parâmetro = 10)	97,19	0,973	0,972	0,972
Naive Bayes I (parâmetro = false)	96,62	0,967	0,966	0,966
Naive Bayes II (parâmetro = true)	98,87	0,989	0,989	0,989
KNN I (parâmetro = 1)	94,94	0,953	0,949	0,949
KNN II (parâmetro = 20)	97.75	0,978	0,978	0,977

Entretanto, em métodos estatísticos existe um fenômeno chamado overfitting [Hawkins 2004], em que um modelo estatístico se ajusta muito bem ao conjunto de dados utilizado como treinamento e teste, mas se mostra ineficaz para prever novos resultados. Um modelo sobre ajustado apresenta alta precisão quando testado com seu conjunto de dados, porém não é uma boa representação da realidade e por isso deve ser evitado.

Com o objetivo de averiguar se os resultados obtidos se tratavam de overfitting, realizou-se uma comparação entre as taxas de erro de treinamento e as taxas de erro de teste, como demonstra a Tabela 27. Dado que as taxas: *Root mean squared error e Mean absolute error* são baixas, descartou-se a possibilidade.

Tabela 27. Comparativo de validação

Classificador	*Root mean squared error*	*Mean absolute error*
RandomForest I (parâmetro = 10)	0,1447	0,0648
RandomForest II (parâmetro = 100)	0,1304	0,0655
RandomForest III (parâmetro = 5)	0,143	0,0788
RandomForest IV (parâmetro = 10)	0,1524	0,0799
Naive Bayes I (parâmetro = false)	0,1294	0,0217
Naive Bayes II (parâmetro = true)	0,092	0,0146
KNN I (parâmetro = 1)	0,1821	0,0413
KNN II (parâmetro = 20)	0,148	0,0629

A melhor acurácia obtida foi com a técnica *Naive Bayes* II (parâmetro = true) a qual, 176 instâncias foram classificadas corretamente. Além disso, para esta técnica, foram obtidas as menores taxas de: *Root mean squared error e Mean absolute error.* Tal resultado foi obtido usando o parâmetro de discretização supervisionada ativo. No entanto, a pior acurácia foi obtida para o modelo KNN I (parâmetro = 1) que classificou corretamente 169 instâncias. Percentualmente, existe uma diferença de aproximadamente 4% entre a melhor acurácia e a pior. Demonstrando que a base não é desafiadora para o processo de classificação, uma vez que para as diferentes técnicas e variações nos parâmetros de entrada a diferença máxima foi pequena.

4. Regras de Associação

As regras de Associação têm como premissa básica encontrar elementos que implicam na presença de outros elementos em uma mesma transação, ou seja, encontrar relacionamentos ou padrões frequentes entre conjuntos de dados. Uma regra de associação tem a forma $A \Rightarrow B$. Onde A e B são conjuntos de pares de valor de atributo. A é chamado de antecedente e B de consequente da regra. Uma regra significa que quando A acontece, B também acontece com uma determinada frequência.

As regras de associação são geralmente analisadas usando um valor mínimo de suporte. A métrica de suporte de uma regra é a proporção de quantas vezes A e B

aparecem juntos, na mesma instância, no conjunto de dados. É comum usar o suporte de um item. Nesse caso, significa a frequência desse item no conjunto de dados. Outra métrica importante é a confiança que representa a probabilidade do consequente, visto que o precedente aconteceu. Por fim, o *lift* diz quantas vezes o antecedente eleva a chance do consequente acontecer. Por exemplo, um aumento = 5,0 significa que quando o antecedente acontece, o consequente têm cinco vezes mais probabilidade de acontecer.

4.1. Apriori

O Apriori foi criado em 1994 por R. Agrawal e R. Srikant e, tem como objetivo encontrar conjuntos de itens frequentes em um conjunto de dados. O nome do algoritmo é Apriori porque ele utiliza o conhecimento prévio das propriedades frequentes do conjunto de itens. Aplica-se uma abordagem iterativa ou pesquisa por nível, onde conjuntos de itens k-frequentes são usados para encontrar conjuntos de itens k + 1.

Para melhorar a eficiência da geração por nível de conjuntos de itens frequentes, uma propriedade importante é usada, ela ajuda a reduzir o espaço de pesquisa. Tal propriedade, assume que todos os subconjuntos de um conjunto de itens frequente devem ser frequentes. Se um conjunto de itens for infrequente, todos os seus super conjuntos serão infrequentes.

Visto isso, para a aplicação do algoritmo foi-se necessário realizar a discretização dos atributos da base. Inicialmente, utilizou-se as medidas *default* de execução para o algoritmo, entretanto, como poucas regras foram geradas, o Suporte mínimo foi alterado para 0.01 e um número máximo de 5k regras. A seguir, os pontos mais importantes serão discutidos.

A Tabela 28 exemplifica algumas das regras que foram encontradas quando tem-se como consequente, a Classe 3. O suporte das regras encontradas possui uma variação entre 0.07 e 0.1. Antecedentes como: *hue='(-inf-0.603]'* e *od280/od315ofdilutedwines='(-inf-1.543]'* aumentam em 3.71 as chances da Classe 3 acontecer. Além disso, combinações como: *magnesium='(88.4-97.6]' flavanoids='(-inf-0.814]'* e *flavanoids='(-inf-0.814]' proline='(558.4-698.6]'* aumentam em 3.71 as chances da Classe 3 acontecer.

Tabela 28. Regras com antecedente classe 3

Antecedente	Consequente	Sup.	Conf.	Lift	Itens
hue='(-inf-0.603]'	classvalues=3	0.1	1.0	3.71	18
flavanoids='(-inf-0.814]' od280/od315ofdilutedwines='(1.543-1.816]'	classvalues=3	0.1	1.0	3.71	18
alcalinityofash='(18.36-20.3]' flavanoids='(-inf-0.814]'	classvalues=3	0.08	1.0	3.71	14
magnesium='(88.4-97.6]' flavanoids='(-inf-0.814]'	classvalues=3	0.08	1.0	3.71	14
flavanoids='(-inf-0.814]' proline='(558.4-698.6]'	classvalues=3	0.08	1.0	3.71	14
od280/od315ofdilutedwines='(-inf-1.543]'	classvalues=3	0.07	1.0	3.71	12
ash='(2.295-2.482]' flavanoids='(-inf-0.814]'	classvalues=3	0.07	1.0	3.71	12

flavanoids='(-inf-0.814]' proanthocyanins='(0.727-1.044]'	classvalues=3	0.07	1.0	3.71	12
totalphenols='(1.27-1.56]' flavanoids='(-inf-0.814]'	classvalues=3	0.07	1.0	3.71	12
totalphenols='(1.56-1.85]' flavanoids='(-inf-0.814]'	classvalues=3	0.07	1.0	3.71	12

A Tabela 29 mostra algumas das regras que foram encontradas quando tem-se como consequente, a Classe 2. O suporte das regras encontradas possui uma variação entre 0.07 e 0.11. Antecedentes como: *alcohol='(11.79-12.17]'* e *colorintensity='(-inf-2.452]'* aumentam em 2.61 as chances da Classe 2 acontecer. Além disso, combinações como: *magnesium='(79.2-88.4]' colorintensity='(2.452-3.624]' e malicacid='(1.246-1.752]' magnesium='(79.2-88.4]'* aumentam em 2.61 as chances da Classe 2 acontecer.

Tabela 29. Regras com antecedente classe 2

Antecedente	Consequente	Sup	Conf.	Lift	Itens
alcohol='(11.79-12.17]'	classvalues=2	0.11	1.0	2.51	18
magnesium='(79.2-88.4]' colorintensity='(2.452-3.624]'	classvalues=2	0.1	1.0	2.51	18
colorintensity='(-inf-2.452]'	classvalues=2	0.09	1.0	2.51	16
malicacid='(1.246-1.752]' magnesium='(79.2-88.4]'	classvalues=2	0.09	1.0	2.51	16
malicacid='(1.246-1.752]' colorintensity='(2.452-3.624]'	classvalues=2	0.08	1.0	2.51	14
alcalinityofash='(20.3-22.24]' colorintensity='(2.452-3.624]'	classvalues=2	0.08	1.0	2.51	14
alcohol='(11.79-12.17]' colorintensity='(2.452-3.624]'	classvalues=2	0.07	1.0	2.51	12
ash='(2.108-2.295]' colorintensity='(2.452-3.624]'	classvalues=2	0.07	1.0	2.51	12
magnesium='(79.2-88.4]' proanthocyanins='(1.361-1.678]'	classvalues=2	0.07	1.0	2.51	12
magnesium='(79.2-88.4]' od280/od315ofdilutedwines='(2.635-2.908]'	classvalues=2	0.07	1.0	2.51	12
colorintensity='(2.452-3.624]' od280/od315ofdilutedwines='(2.908-3.181]'	classvalues=2	0.07	1.0	2.51	12

Para finalizar, a Tabela 30 mostra algumas das regras que foram encontradas quando tem-se como consequente, a Classe 1. O suporte das regras encontradas possui uma variação entre 0.04 e 0.08. Antecedentes como: *proline='(979-1119.2]', proline='(1119.2-1259.4]', , proline='(1259.4-1399.6]' e flavanoids='(3.184-3.658]'*

aumentam em 3.02 as chances da Classe 1 acontecer. Além disso, combinações como: *totalphenols='(2.72-3.01]' hue='(0.972-1.095]' e malicacid='(1.752-2.258]' flavanoids='(2.71-3.184]'* aumentam em 2.61 as chances da Classe 1 acontecer.

Tabela 30. Regras com antecedente classe 1

Antecedente	Consequente	Sup	Conf.	Lift	Itens
flavanoids='(3.184-3.658]'	classvalues=1	0.08	1.0	03.02	14
proline='(1259.4-1399.6]'	classvalues=1	0.07	1.0	03.02	12
totalphenols='(2.72-3.01]' hue='(0.972-1.095]'	classvalues=1	0.06	1.0	03.02	11
malicacid='(1.752-2.258]' flavanoids='(2.71-3.184]'	classvalues=1	0.06	1.0	03.02	11
magnesium='(97.6-106.8]' proline='(979-1119.2]'	classvalues=1	0.06	1.0	03.02	11
flavanoids='(2.71-3.184]' hue='(0.972-1.095]'	classvalues=1	0.06	1.0	03.02	11
malicacid='(1.752-2.258]' totalphenols='(2.72-3.01]'	classvalues=1	0.05	1.0	03.02	9
alcalinityofash='(16.42-18.36]' proline='(1259.4-1399.6]'	classvalues=1	0.05	1.0	03.02	9
colorintensity='(3.624-4.796]' proline='(979-1119.2]'	classvalues=1	0.05	1.0	03.02	9
proline='(1119.2-1259.4]'	classvalues=1	0.04	1.0	03.02	7

Existem ainda, atributos que influenciam na ocorrência de outros atributos. A Tabela 31 mostra algumas das regras que foram encontradas, o resultado por completo pode ser encontrado no link: https://docs.google.com/spreadsheets/d/12tmmmfiY9TfY71IM0VRN55qQLzRG5PqHyhBJgRPe0Gc/edit?usp=sharing. Para essas regras, verificou-se o Lift que varia entre 35.6 e 25.43, ou seja, quando o antecedente acontece, o consequente têm entre 35 vezes e 25 vezes mais probabilidade de acontecer. Além disso, o suporte para os itens é baixo, 0.02/0.03, assim temos uma correlação forte para suporte baixo.

Tabela 31. Regras com outros atributos

Antecedente	Consequente	Sup.	Conf.	Lift
totalphenols='(2.72-3.01]' proline='(-inf-418.2]'	magnesium='(79.2-88.4]' flavanoids='(2.71-3.184]'	0.02	1.0	35.6
ash='(2.295-2.482]' proanthocyanins='(0.727-1.044]' od280/od315ofdilutedwines='(-inf-1.543]'	alcohol='(12.55-12.93]' flavanoids='(0.814-1.288]'	0.02	1.0	35.6

alcohol='(12.55-12.93]' ash='(2.295-2.482]' flavanoids='(0.814-1.288]'	proanthocyanins='(0.727-1.044]' od280/od315ofdilutedwines='(-inf-1.543]'	0.02	1.0	35.6
alcohol='(12.55-12.93]' flavanoids='(0.814-1.288]' classvalues=3	proanthocyanins='(0.727-1.044]' od280/od315ofdilutedwines='(-inf-1.543]'	0.02	1.0	35.6
totalphenols='(2.72-3.01]' proline='(-inf-418.2]' classvalues=2	magnesium='(79.2-88.4]' flavanoids='(2.71-3.184]'	0.02	1.0	35.6
alcohol='(11.79-12.17]' hue='(0.972-1.095]'	ash='(2.295-2.482]' proanthocyanins='(1.361-1.678]' classvalues=2	0.02	1.0	29.67
alcohol='(12.55-12.93]' proanthocyanins='(0.727-1.044]' od280/od315ofdilutedwines='(-inf-1.543]'	ash='(2.295-2.482]' flavanoids='(0.814-1.288]' classvalues=3	0.02	1.0	29.67
alcohol='(12.55-12.93]' colorintensity='(4.796-5.968]' proline='(558.4-698.6]'	magnesium='(97.6-106.8]' proanthocyanins='(0.727-1.044]' classvalues=3	0.02	1.0	29.67
alcohol='(13.69-14.07]' malicacid='(1.246-1.752]' proline='(979-1119.2]'	ash='(2.108-2.295]' alcalinityofash='(14.48-16.42]' classvalues=1	0.02	1.0	29.67
alcohol='(12.55-12.93]' colorintensity='(4.796-5.968]' classvalues=3	magnesium='(97.6-106.8]' proanthocyanins='(0.727-1.044]'	0.03	1.0	25.43
totalphenols='(2.72-3.01]' nonflavanoidphenols='(0.236-0.289]'	proanthocyanins='(1.995-2.312]' colorintensity='(4.796-5.968]'	0.02	1.0	25.43
alcohol='(12.55-12.93]' proanthocyanins='(0.727-1.044]' od280/od315ofdilutedwines='(-inf-1.543]'	ash='(2.295-2.482]' flavanoids='(0.814-1.288]'	0.02	1.0	25.43
alcohol='(12.55-12.93]' flavanoids='(0.814-1.288]' classvalues=3	ash='(2.295-2.482]' od280/od315ofdilutedwines='(-inf-1.543]'	0.02	1.0	25.43
alcohol='(12.55-12.93]' colorintensity='(4.796-5.968]' proline='(558.4-698.6]'	magnesium='(97.6-106.8]' proanthocyanins='(0.727-1.044]'	0.02	1.0	25.43

5. Clusterização

A análise de cluster ou simplesmente clusterização é o processo de particionar um conjunto de objetos, dados ou observações em subconjuntos. Cada subconjunto é denominado como cluster, de forma que os objetos em um cluster são semelhantes entre si e dissemelhantes entre os objetos de outros clusters. O conjunto de clusters resultante de uma análise pode ser referido como clusterização. Nesse contexto, diferentes métodos de agrupamento podem gerar diferentes agrupamentos no mesmo conjunto de dados. O agrupamento é útil, uma vez que pode levar à descoberta de grupos previamente desconhecidos dentro dos dados [Han and Kamber 2012] .

Segundo Han and Kamber, a clusterização tem sido amplamente utilizada em muitas aplicações, como, por exemplo: *business intelligence*, reconhecimento de padrões de imagem, pesquisa na Web, biologia e segurança. Em *business intelligence*, o clustering pode ser usado para organizar um grande número de clientes em grupos, onde os clientes dentro de um grupo compartilham fortes características semelhantes. Isso facilita o desenvolvimento de estratégias de negócios para uma gestão aprimorada do relacionamento com o cliente.

Como uma função de mineração de dados, a análise de cluster pode ser usada como uma ferramenta independente para obter *insights* sobre a distribuição de dados, observar as características de cada cluster e se concentrar em um determinado conjunto de clusters para análise posterior [Han and Kamber 2012]**.**

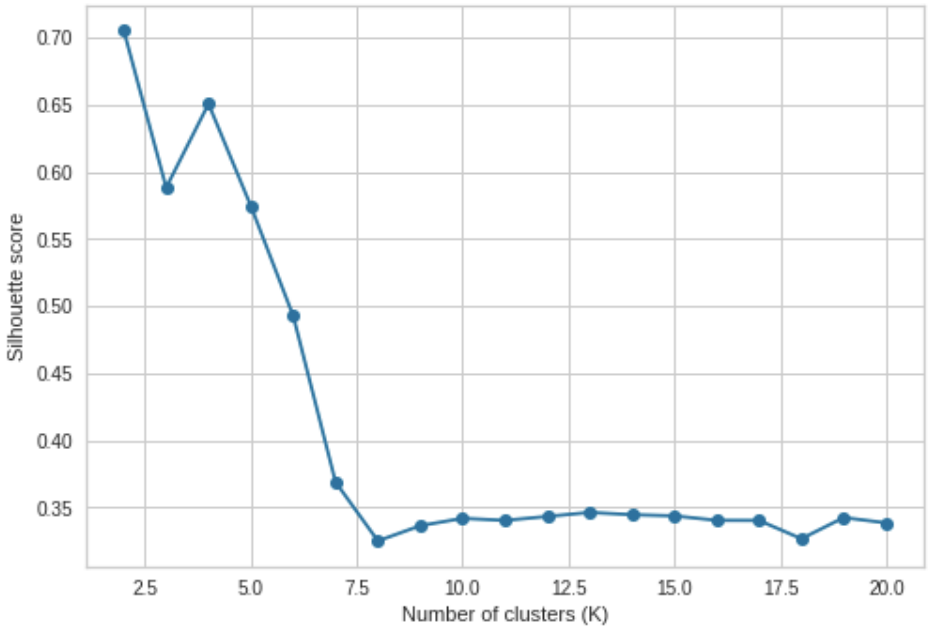
5.1. K-Means

Na literatura, o K-Means [Macqueen 1967] é um algoritmo de agrupamento típico, baseado na noção de vizinhança, que visa particionar N entradas em K clusters (dado o parâmetro K), com base na distância entre suas instâncias.

Segundo Pham et al., o algoritmo K-means é um algoritmo popular de clusterização. Para usá-lo, é necessário que o número de clusters nos dados seja pré-especificado. Encontrar o número apropriado de clusters para um determinado conjunto de dados é geralmente um processo de tentativa e erro que se torna mais difícil pela natureza subjetiva de decidir o que constitui um agrupamento "correto".

Diante disso, executou-se o algoritmo K-means com K variando de 2 a 20. Para cada execução, calculou-se a média dos coeficientes de silhueta. A Figura 4 contém os resultados obtidos a partir desta execução.

Figura 4. Silhueta x número de clusters



Segundo a Figura 4, pode-se observar que os maiores coeficientes encontrados foram para K = 2 e K = 4. As Figuras 5 e 6 mostram os coeficientes de silhueta de cada um dos 13 atributos da base de dados.

Para K= 2, 108 instâncias, ou seja, 61% dos itens, estão contidas no cluster 0 e 70 instâncias, ou seja, 36% dos itens estão contidas no cluster 1. Analisando a Figura 5 onde cada centróide dos atributos e seus desvios padrão são mostrados, pode-se observar que o atributo *NonFlavanoidPhenols* caracteriza os clusters, uma vez que seu desvio padrão é baixo. Atributos como, *Proline, Magnesium e AlcalinityoFash* não foram informações que caracterizam fortemente, uma vez que seu desvio padrão é alto.

Além disso, observa-se que, para o cluster 0, os atributos: *Alcohol, Magnesium, TotalPhenols, Flavanoids, Proanthocyanins, Hue, OD280 e Proline* possuírem os valores do centróide maiores que os centróides da base inteira. Já para o cluster 1, os atributos são: *Malicacid, Ash, Alcalinityofash, NonFlavanoidPhenols e ColorIntensity.*

Figura 5. Centróides para K = 2

```
Final cluster centroids:
                                        Cluster#
Attribute                  Full Data           0           1
                             (178.0)     (108.0)      (70.0)
=============================================================
alcohol                       13.0006     13.0809     12.8767
                             +/-0.8118   +/-0.8941   +/-0.6522

malicacid                      2.3363      1.9249      2.9711
                             +/-1.1171   +/-0.8455   +/-1.1913

ash                            2.3665      2.3443      2.4009
                             +/-0.2743   +/-0.2912   +/-0.2442

alcalinityofash               19.4949     18.5296     20.9843
                             +/-3.3396   +/-3.2566   +/-2.9076

magnesium                     99.7416    100.9352        97.9
                            +/-14.2825  +/-15.6111  +/-11.8189

totalphenols                   2.2951      2.6738      1.7109
                             +/-0.6259   +/-0.4492   +/-0.3481

flavanoids                     2.0293      2.6898      1.0101
                             +/-0.9989   +/-0.6219   +/-0.4722

nonflavanoidphenols            0.3619      0.3008       0.456
                             +/-0.1245   +/-0.0789   +/-0.1235

proanthocyanins                1.5909      1.8576      1.1794
                             +/-0.5724   +/-0.4946   +/-0.4217

colorintensity                 5.0581      4.4097      6.0584
                             +/-2.3183   +/-1.6628   +/-2.7975

hue                            0.9574      1.0671      0.7882
                             +/-0.2286   +/-0.1654   +/-0.2091

od280/od315ofdilutedwines      2.6117      3.0891      1.8751
                               +/-0.71   +/-0.3572   +/-0.4338

proline                      746.8933    844.6852    596.0143
                           +/-314.9075 +/-358.3302 +/-131.1767
```

Para K= 4, 48 instâncias, ou seja, 27% dos itens, estão contidas no cluster 0. 31 instâncias, ou seja, 17% dos itens, estão contidas no cluster 1. 44 instâncias, ou seja, 25% dos itens, estão contidas no cluster 2 e 55 instâncias, ou seja, 31% dos itens, estão contidas no cluster 3. Analisando a Figura 6 onde cada centróide dos atributos e seus desvios padrão são mostrados, pode-se observar que o atributo *Hue* caracteriza os clusters, uma vez que seu desvio padrão é baixo. Atributos como, *Proline e Magnesium* não foram informações que caracterizam fortemente, uma vez que seu desvio padrão é alto.

Além disso, os clusters 1, 2, se assemelham com relação aos atributos: *Alcohol*, com centróides de aproximadamente 12; *Ash*, com centróides de aproximadamente 2.2; *Alcalinityofash*, centróides de aproximadamente 20.1; *Hue* centróides de aproximadamente 1.0.

Figura 6. Centróides para K = 4

```
Final cluster centroids:
                                        Cluster#
Attribute                  Full Data            0           1           2           3
                             (178.0)       (48.0)      (31.0)      (44.0)      (55.0)
=============================================================================================
alcohol                     13.0006       13.165      12.2381     12.3711     13.7905
                           +/-0.8118     +/-0.522    +/-0.5811   +/-0.5288   +/-0.4434

malicacid                    2.3363       3.3883      1.7503      2.1082      1.9311
                           +/-1.1171     +/-1.0455   +/-0.7523   +/-1.1586   +/-0.6045

ash                          2.3665       2.4385      2.2655      2.2752      2.4336
                           +/-0.2743     +/-0.1835   +/-0.2979   +/-0.3658   +/-0.1993

alcalinityofash             19.4949      21.5208     20.1774     20.1932     16.7836
                           +/-3.3396     +/-2.1879   +/-3.5006   +/-3.2474   +/-2.3204

magnesium                   99.7416      99.3125     89.6129     99.5909     105.9455
                          +/-14.2825    +/-10.8905   +/-9.9051  +/-19.5142  +/-10.4588

totalphenols                 2.2951       1.6715       1.901      2.5518      2.8562
                           +/-0.6259     +/-0.3538   +/-0.3717   +/-0.4631   +/-0.3437

flavanoids                   2.0293       0.8025        1.58      2.4582        3.01
                           +/-0.9989     +/-0.3142   +/-0.4168   +/-0.6605   +/-0.3966

nonflavanoidphenols          0.3619       0.4477      0.4539      0.2993      0.2851
                           +/-0.1245     +/-0.1244   +/-0.1048   +/-0.0881   +/-0.0664

proanthocyanins              1.5909       1.1613      1.2361      1.9191      1.9033
                           +/-0.5724     +/-0.4141   +/-0.4112   +/-0.5327   +/-0.4255

colorintensity               5.0581       7.3827      3.0648       3.217      5.6256
                           +/-2.3183     +/-2.3245   +/-0.8899   +/-1.0046   +/-1.2231

hue                          0.9574       0.6846      1.0683      1.0411      1.0662
                           +/-0.2286     +/-0.1163    +/-0.216   +/-0.1916   +/-0.1156

od280/od315ofdilutedwines    2.6117        1.694      2.4419      3.0502      3.1575
                             +/-0.71     +/-0.2749   +/-0.4948   +/-0.3402   +/-0.3648

proline                    746.8933     627.0833    524.6129    540.2727    1142.0364
                         +/-314.9075   +/-116.2391  +/-123.9878 +/-186.6137  +/-205.124
```

Uma vez que a base utilizada possuía a atributo classe, verificou-se os resultados obtidos pelas variações de K entre 2 e 20 coincidem com a distribuição das classes. A Tabela 32 apresenta os números e os percentuais de instâncias clusterizadas de forma incorreta. Observa-se que os menores valores foram encontrados para K = 3, K = 4 e K= 6.

Tabela 32. Instâncias clusterizadas de forma incorreta

Número de K	Número de instâncias	Percentual
2	71.0	39.8876 %
3	10.0	5.618 %
4	36.0	20.2247 %
5	63.0	35.3933 %
6	57.0	32.0225 %

7	73.0	41.0112 %
8	75.0	42.1348 %
9	89.0	50 %
10	75.0	42.1348 %
11	96.0	53.9326 %
12	105.0	58.9888 %
13	107.0	60.1124 %
14	102.0	57.3034 %
15	104.0	58.427 %
16	114.0	64.0449 %
17	112.0	62.9213 %
18	124.0	69.6629 %
19	123.0	69.1011 %
20	121.0	67.9775 %

5.2. DBSCAN

DBSCAN é um algoritmo de clusterização que, diferentemente de K-means, se baseia no conceito de vizinhança, e na noção de densidade [Sander et al. 1998]. Para encontrar a densidade de uma instância, ele verifica o número de outras instâncias próximas a ela. Desta forma, existem dois parâmetros principais para este algoritmo: ε ou eps o qual é o raio considerado ao pesquisar por instâncias próximas a outra instância. minP ou ts, o qual é a quantidade mínima de instâncias que devem existir perto de outra para que uma região densa seja identificada e agrupada.

A Tabela 33 mostra as execuções realizadas com seus parâmetros de entrada: *Eps* e *minP*, além do números de clusters gerados, instâncias não clusterizadas e as instâncias que foram clusterizadas de forma incorreta com relação a distribuição original da base. Para todas as execuções foi utilizada a distância Euclidiana.

Tabela 33. Execuções realizadas com Eps e minP

Eps	minP	Número de Clusters	Instâncias não Clusterizadas	Clusterização Incorreta
0.1	6	0	178	-
0.2	6	0	178	-
0.3	6	0	178	-
0.36	6	4	145	11
0.4	2	5	52	37

0.4	6	4	86	26
0.4	4	4	63	47
0.4	5	4	78	28
0.41	6	4	79	27
0.45	6	2	55	35
0.47	6	2	42	41
0.49	6	2	32	45
0.5	3	2	22	52
0.5	4	2	23	51
0.5	5	2	23	51
0.5	6	2	29	47
0.5	7	2	29	47
0.51	6	2	25	49

A partir dos resultados obtidos, nota-se que para a variável *eps* para valores menores que 0.3 clusters não foram formados. Além disso, nenhuma das execuções gerou o número de clusters da distribuição original da base, ou seja, 3 classes. Para a variável eps= 0.5, minP variando entre 3 e 5, dois clusters foram gerados, e obteve-se os menores valores para as instâncias não classificadas. A Figura 7 mostra a distribuição dos valores citados.

Figura 7. Clusters gerados para eps=0.5

```
Clustered Instances

0      105 ( 67%)
1       51 ( 33%)

Unclustered instances : 22

Class attribute: classvalues
Classes to Clusters:

  0  1  <-- assigned to cluster
 58  0 | 1
 47  5 | 2
  0 46 | 3

Cluster 0 <-- 1
Cluster 1 <-- 3
```

```
Clustered Instances

0      104 ( 67%)
1       51 ( 33%)

Unclustered instances : 23

Class attribute: classvalues
Classes to Clusters:

  0  1  <-- assigned to cluster
 58  0 | 1
 46  5 | 2
  0 46 | 3

Cluster 0 <-- 1
Cluster 1 <-- 3
```

```
Clustered Instances

0      104 ( 67%)
1       51 ( 33%)

Unclustered instances : 23

Class attribute: classvalues
Classes to Clusters:

  0  1  <-- assigned to cluster
 58  0 | 1
 46  5 | 2
  0 46 | 3

Cluster 0 <-- 1
Cluster 1 <-- 3
```

6. Auto-Weka

Auto-Weka é um pacote Weka que emprega técnicas de otimização para realizar o ajuste automático de parâmetros de algoritmos de classificação e seleção de modelo. Pode ser instalado a partir do gerenciador de pacotes uma vez que não está disponível por padrão na ferramenta.

Segundo o guia[3] ao utilizar o Auto-WEKA como um classificador normal, é importante selecionar a opção Teste *"Use training set"*. O Auto-WEKA realiza internamente uma avaliação estatisticamente rigorosa (validação cruzada de 10 vezes) e não requer a divisão externa em conjuntos de treinamento e teste que o WEKA fornece. Selecionar outra opção não melhora a qualidade do resultado e fará com que o Auto-WEKA demore muito mais. Assim, utilizou-se a configuração recomendada por padrão.

Após a execução, foi relatado o melhor modelo encontrado. Para a base utilizada, o melhor modelo encontrado por Auto-Weka foi o Simple Logistic com acurácia de 100%.

Tabela 34. Resumo da validação para Simple Logistic

Correctly Classified Instances	178 (100 %)
Incorrectly Classified Instances	0 (0 %)
Kappa statistic	1
Mean absolute error	0.0123
Root mean squared error	0.0411
Relative absolute error	2.7972 %
Root relative squared error	8.7701 %

Tabela 35. Detalhamento de acurácia para Simple Logistic

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	1,000	0,000	1,000	1,000	1,000	1,000	1,000	1,000	1
	1,000	0,000	1,000	1,000	1,000	1,000	1,000	1,000	2
	1,000	0,000	1,000	1,000	1,000	1,000	1,000	1,000	3
Média ponderada	1,000	0,000	1,000	1,000	1,000	1,000	1,000	1,000	

Tabela 36. Matriz de Confusão para Simple Logistic

a	b	c	Classificado como
59	0	0	a=1
0	71	0	b=2
0	0	48	c=3

[3] http://www.cs.ubc.ca/labs/beta/Projects/autoweka/manual.pdf

Referências

Dua, D. and Graff, C. (2019). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.

Bramer, M. (2016). Principles of data mining. 3rd edition. 2016 ed. New York, NY: Springer London.

Han, J. and Kamber, M. (2012). Data mining: concepts and techniques. 3rd ed ed. Burlington, MA: Elsevier.

Hawkins, D. M. (1 jan 2004). The Problem of Overfitting. Journal of Chemical Information and Computer Sciences, v. 44, n. 1, p. 1–12.

Kesavaraj, G. and Sukumaran, S. (2013). A study on classification techniques in data mining. In 2013 Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT).

Macqueen, J. (1967). Some methods for classification and analysis of multivariate observations. In In 5-th Berkeley Symposium on Mathematical Statistics and Probability.

Pham, D. T., Dimov, S. S. and Nguyen, C. D. (2005). Selection of K in K-means clustering. Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science, v. 219, n. 1, p. 103–119.

Russell, S. J., Norvig, P. and Davis, E. (2010). Artificial intelligence: a modern approach. 3rd ed ed. Upper Saddle River: Prentice Hall.

Sander, J., Ester, M., Kriegel, H.-P. and Xu, X. (1 jun 1998). Density-Based Clustering in Spatial Databases: The Algorithm GDBSCAN and Its Applications. Data Mining and Knowledge Discovery, v. 2, n. 2, p. 169–194.

Simon, P. (2015). Too Big to Ignore: The Business Case for Big Data. John Wiley & Sons.

Winston, P. H. (1992). Artificial intelligence. 3rd ed ed. Reading, Mass: Addison-Wesley Pub. Co.