

Data Science Algorithms



Cruzeiro do Sul Virtual
Educação a distância

Material Teórico



Estudo de Caso com o Algoritmo *k-means*

Responsável pelo Conteúdo:

Prof. Dr. Alberto Messias

Revisão Textual:

Prof.^a Dr.^a Selma Aparecida Cesarin

UNIDADE

Estudo de Caso com o Algoritmo *k-means*



- *Kmeans.*



OBJETIVO DE APRENDIZADO

- Estender o estudo de caso das empresas de cobrança utilizando o algoritmo *Kmeans*, o foco será a execução do *Kmeans* e a análise dos *clusters* gerados, bem como o decaimento do erro RMS dos modelos gerados para a mesma base de dados anterior.



Orientações de estudo

Para que o conteúdo desta Disciplina seja bem aproveitado e haja maior aplicabilidade na sua formação acadêmica e atuação profissional, siga algumas recomendações básicas:

Determine um horário fixo para estudar.

Mantenha o foco! Evite se distrair com as redes sociais.

Procure manter contato com seus colegas e tutores para trocar ideias! Isso amplia a aprendizagem.

Seja original! Nunca plágie trabalhos.

Aproveite as indicações de Material Complementar.

Conserve seu material e local de estudos sempre organizados.

Não se esqueça de se alimentar e de se manter hidratado.

Assim:

- ✓ Organize seus estudos de maneira que passem a fazer parte da sua rotina. Por exemplo, você poderá determinar um dia e horário fixos como seu “momento do estudo”;
- ✓ Procure se alimentar e se hidratar quando for estudar; lembre-se de que uma alimentação saudável pode proporcionar melhor aproveitamento do estudo;
- ✓ No material de cada Unidade, há leituras indicadas e, entre elas, artigos científicos, livros, vídeos e sites para aprofundar os conhecimentos adquiridos ao longo da Unidade. Além disso, você também encontrará sugestões de conteúdo extra no item **Material Complementar**, que ampliarão sua interpretação e auxiliarão no pleno entendimento dos temas abordados;
- ✓ Após o contato com o conteúdo proposto, participe dos debates mediados em fóruns de discussão, pois irão auxiliar a verificar o quanto você absorveu de conhecimento, além de propiciar o contato com seus colegas e tutores, o que se apresenta como rico espaço de troca de ideias e de aprendizagem.



Kmeans

Considerando o mesmo estudo de caso da unidade anterior, as empresas de cobrança que precisam otimizar os seus resultados financeiros utilizando a base de dados histórica de clientes que fecham acordos, estenderemos o estudo de caso com o experimento usando o algoritmo *Kmeans*.

Para os experimentos nessa unidade utilizaremos apenas o cliente denominado A, vocês têm acesso à base de dados do cliente A para reproduzirem os experimentos aqui desenvolvimento utilizando o software *Weka*.

Segue a descrição dos atributos que encontrarão na base de dados do cliente A:

- A base de dados possui os seguintes atributos:
 - » **Idade:** Idade do devedor;
 - » **Atraso:** Quantidade de dias de Atraso;
 - » **Valor:** Valor do débito em atraso;
 - » **Contato:** Se houve contato pela central;
 - » **Efetivo:** quantidade de contatos efetivados pela central;
 - » **Acordo:** se foi fechado o acordo de pagamento ou não, atributo objetivo da análise.

Para a melhor execução do algoritmo no software *Weka*, optou-se por transformar todos os atributos em numéricos e com faixas, de modo a facilitar o entendimento do modelo e otimizar a execução do algoritmo em escala.

Seguem as definições das classes e faixas criadas com os respectivos códigos numéricos utilizados:

Codificação para as faixas etárias dos clientes devedores:

- **Código 0:** De 0 a 25 anos;
- **Código 1:** De 26 a 35 anos;
- **Código 2:** De 36 a 45 anos;
- **Código 4:** Maior que 45 anos.

Codificação para as faixas de valores das dívidas:

- **Código 0:** De 0 a 200 reais;
- **Código 1:** De 200 a 500 reais;
- **Código 2:** De 500 a 1000 reais;
- **Código 5:** Maior que 1000 reais.

Codificação para as faixas de tempo de atraso:

- **Código 0:** De 0 a 15 dias;
- **Código 1:** De 15 a 30 dias;

- **Código 2:** De 31 a 120 dias;
- **Código 6:** acima de 120 dias.

Para o experimento optou-se por trabalhar com o arquivo no formato CSV, separado pelo caractere “.”. O software Weka suporta arquivos nesse formato, segue a tela na qual o arquivo da base de dados do cliente A é importada para os experimentos.

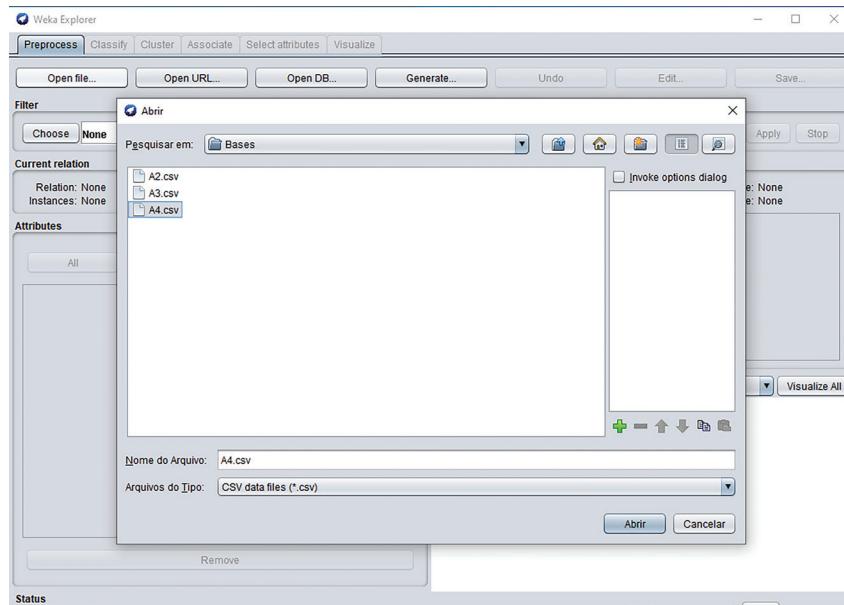


Figura 1 – Importação da base de dados do cliente A

Fonte: Acervo do conteudista

Logo após a importação da base o Weka deverá reconhecer os atributos existentes na base de dados, note que o arquivo possui então os 6 atributos e 4773 instâncias no total. Segue a figura 2 que ilustra os atributos logo após a importação dos dados.

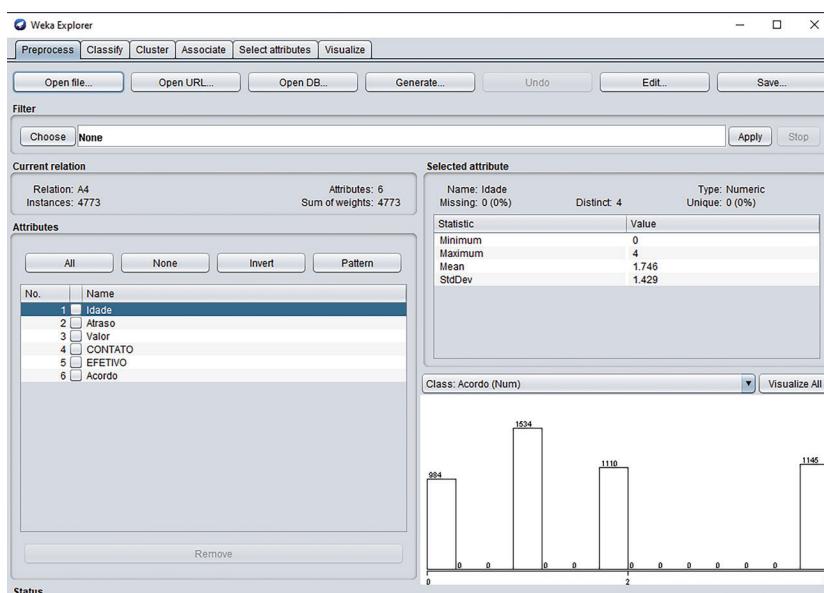


Figura 2 – Tela de importação de dados para o Weka

Fonte: Acervo do conteudista

Após a importação dos dados, devemos ir então para a guia que possui as implementações dos diversos algoritmos, como algoritmos para: classificação, *clustering*, regras de associação, seleção de atributos, e, por fim, a visualização dos resultados. Segue a figura 3, na qual deve-se clicar na aba “cluster”, e por sua vez no algoritmo “*kmeans*”.

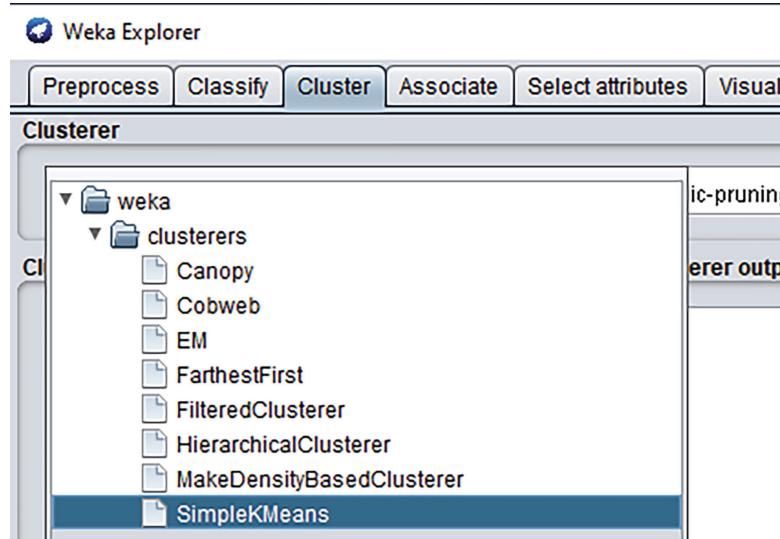


Figura 3 – Tela de seleção de tipo de algoritmo no Weka

Fonte: Acervo do conteudista

Ao selecionar o algoritmo *kmeans* você poderá clicar na linha de parâmetros para a execução do algoritmo, nesse caso, para nosso experimento iremos variar o número de *clusters*, partindo de 1 e indo até 15 de modo a observarmos o decaimento do erro RMS. A Figura 4 ilustra as telas que são exibidas ao clicar na linha, uma opção que pode ser alterada é o tipo de medida de distância utilizada, em nosso caso utilizaremos a distância euclidiana, já pré-selecionada no Weka.

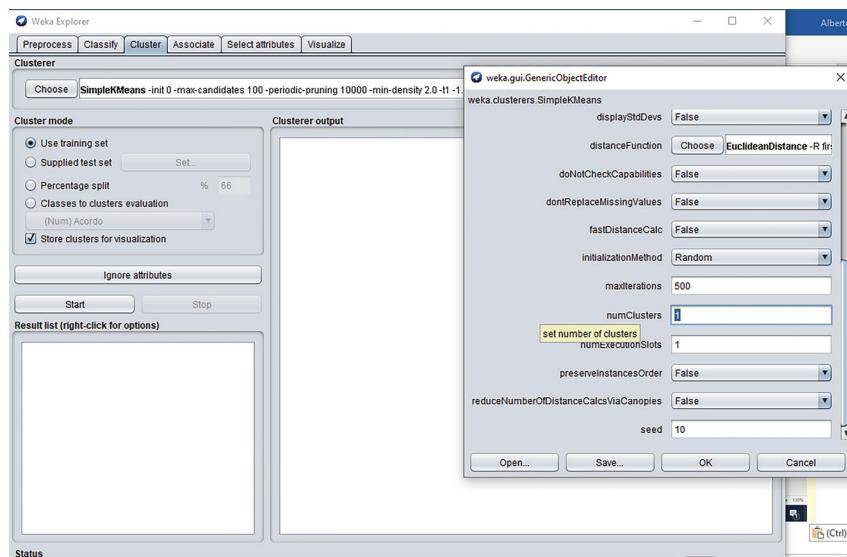
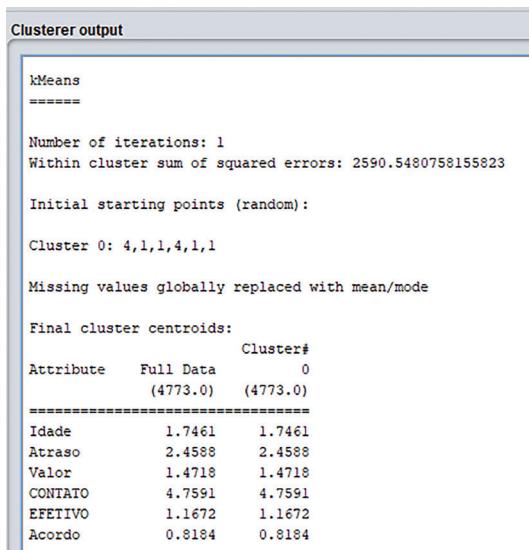


Figura 4 – Alteração de parâmetros para o algoritmo *kmeans*

Fonte: Acervo do conteudista

Ao clicar em “OK” na tela de seleção de parâmetros podemos ir para a execução do algoritmo, nesse caso, clicando no botão “Start”. Ao centro da tela, ou

“Clusterer output” são exibidas as saídas do processamento do algoritmo, conforme pode-se observar na figura 5.



```

Clusterer output

kMeans
=====

Number of iterations: 1
Within cluster sum of squared errors: 2590.5480758155823

Initial starting points (random):

Cluster 0: 4,1,1,4,1,1

Missing values globally replaced with mean/mode

Final cluster centroids:
      Cluster#
Attribute   Full Data      0
              (4773.0) (4773.0)
=====
Idade       1.7461    1.7461
Atraso      2.4588    2.4588
Valor       1.4718    1.4718
CONTATO    4.7591    4.7591
EFETIVO    1.1672    1.1672
Acordo     0.8184    0.8184
  
```

Figura 5 – Saída de execução do algoritmo no Weka

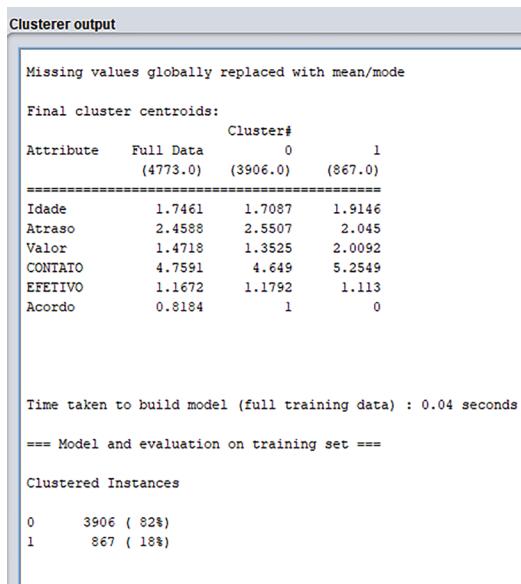
Fonte: Acervo do conteudista

Na saída do algoritmo é importante observarmos o erro RMS que é exibido, bem como a tabela com os atributos de todas as instâncias e em seguida os atributos de cada centroide de cluster e o seu número de instâncias.

Para o modelo com um único cluster a saída de erro está com o valor apresentado na linha:

Within cluster sum of squared errors: 2590.5480758155823

Vamos alterar o número de *clusters* para 2 e observar o resultado do agrupamento. Segue a figura 6 com a saída.



```

Clusterer output

Missing values globally replaced with mean/mode

Final cluster centroids:
      Cluster#
Attribute   Full Data      0      1
              (4773.0) (3906.0) (867.0)
=====
Idade       1.7461    1.7087   1.9146
Atraso      2.4588    2.5507   2.045
Valor       1.4718    1.3525   2.0092
CONTATO    4.7591    4.649    5.2549
EFETIVO    1.1672    1.1792   1.113
Acordo     0.8184     1        0

Time taken to build model (full training data) : 0.04 seconds

== Model and evaluation on training set ==

Clustered Instances

0      3906 ( 82%)
1      867 ( 18%)
  
```

Figura 6 – Saída do algoritmo para 2 *clusters*

Fonte: Acervo do conteudista

Observe que na saída do Weka os *clusters* são claramente divididos em 2 grupos com o atributo “acordo” 0 e 1, ou seja, um agrupamento adequado, com 3906 instâncias com acordo e 867 instâncias sem acordo. É possível verificar graficamente o agrupamento clicando na opção “**visualize cluster assignments**”, conforme a figura 7.

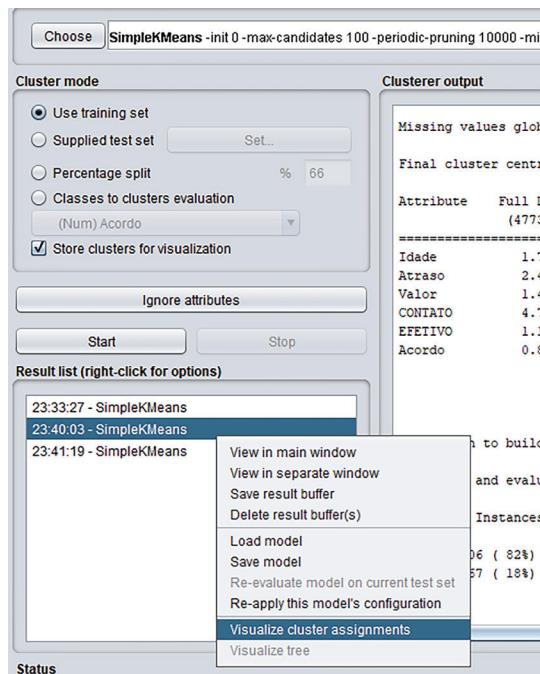


Figura 7 – Entrando na visualização gráfica dos *clusters* gerados

Fonte: Acervo do conteudista

No eixo X deixaremos cada instância da base e no eixo Y o atributo acordo na visualização, foi alterado também o “*Jitter*” apenas para deixar os pontos no gráfico maiores. Segue a figura 8 com o gráfico gerado.

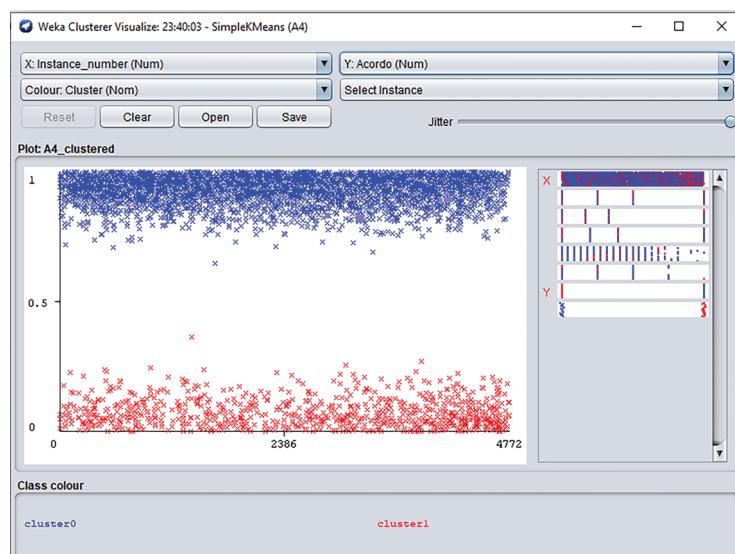


Figura 8 – Visualização gráfica para 2 *clusters*

Fonte: Acervo do conteudista

Note como os grupos estão bem definidos, clientes que não fecham acordo em vermelho e os clientes que fecham acordos em azul e as instâncias nos grupos não se misturam.

Para a nova rodada de experimentos vamos executar o algoritmo com 4 *clusters*, visualmente observamos que há três grupos que fecham acordos e um único que não, conforme se observa na figura 9.

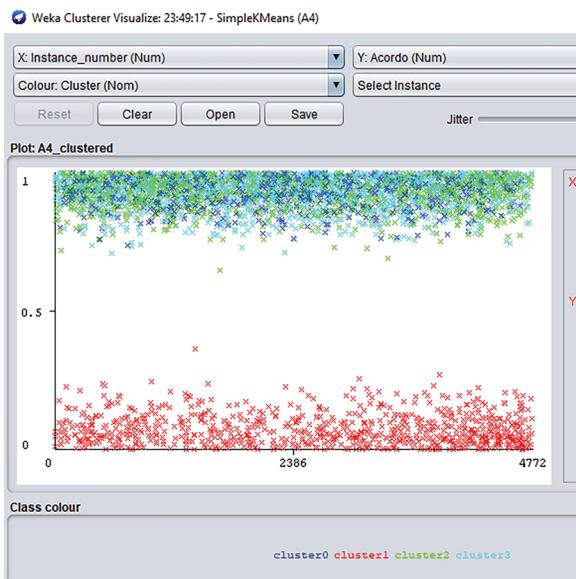


Figura 9 – Visualização gráfica para 4 *clusters*

Fonte: Acervo do conteudista

Note que para 4 *clusters* o algoritmo separa três *clusters* com acordos e um com as instâncias que não fecham acordo.

Vamos executar o experimento para 5 *Clusters*, observando a saída do algoritmo na figura 10.

Clusterer output						
Number of iterations: 16						
Within cluster sum of squared errors: 909.7776150855439						
Initial starting points (random):						
Cluster 0: 4,1,1,4,1,1						
Cluster 1: 4,2,5,10,2,1						
Cluster 2: 4,2,2,2,2,1						
Cluster 3: 1,2,1,2,1,1						
Cluster 4: 1,2,2,4,1,1						
Missing values globally replaced with mean/mode						
Final cluster centroids:						
Attribute	Full Data	Cluster# 0	1	2	3	4
	(4773.0)	(663.0)	(834.0)	(269.0)	(873.0)	(2134.0)
=====						
Idade	1.7461	4	1.8321	4	1.0332	1.0197
Atraso	2.4588	1.181	1.8885	6	6	1.1837
Valor	1.4718	1.9593	2.0731	0.4498	0.3963	1.6542
CONTATO	4.7591	4.3922	5.2842	4.0595	4.7205	4.7718
EFEITIVO	1.1672	1.1825	1.1091	1.223	1.1775	1.1739
Acordo	0.8184	1	0	0.8773	1	1

Figura 10 – Saída do algoritmo em tabelas para 5 *clusters*

Fonte: Acervo do conteudista

Observamos que um dos *clusters* possui o atributo “acordo” com o valor em “0,8773”, o que significa que teria uma mistura de clientes que fecham acordos e que não. Vamos observar visualmente o que ocorre através da figura 11.



Figura 11 – Visualização gráfica com 5 *clusters*

Fonte: Acervo do conteudista

Observe que o *cluster* indicado com a cor verde, com o valor de acordo com “0,8773” se mistura ao *cluster* indicado em vermelho, ou seja, sem acordo, e o maior número de instâncias se misturando aos outros *clusters* que fecham acordo. Note que o valor de “0,87” se aproxima mais do valor “1”, ou seja, um maior número de instâncias misturados aos *clusters* que fecham acordo.

Vamos executar o experimento com 7 *clusters*, note que na saída do algoritmo os grupos de não acordos estão isolados e resta saber qual dos grupos de acordo seria mais rentável investir as ligações no *telemarketing* ativo, segue a figura 12 com a visualização gráfica.

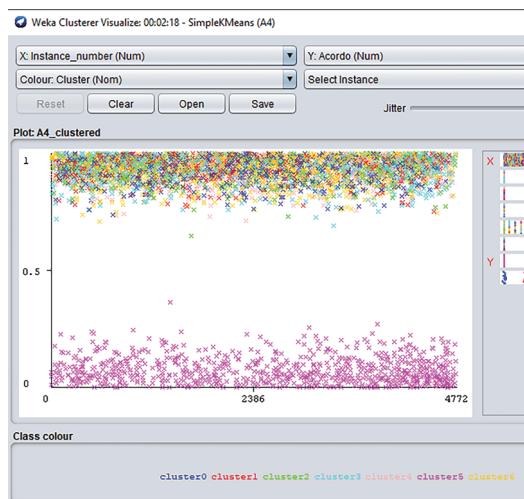


Figura 12 – Visualização gráfica com o experimento para 7 *clusters*

Fonte: Acervo do conteudista

Note que graficamente o *cluster* indicado com a cor rocha é o único *cluster* com o acordo em valor “0”, está completamente isolado, outra observação importante é o decaimento do erro RMS, que para esse experimento ficou em “712.1244016805499”.

Ao executar o experimento com 8 clusters, o algoritmo começa a separar os clientes que não fecham acordos, talvez irrelevantes para a análise, segue a figura 13 com a saída do algoritmo.

Clusterer output											
Number of iterations: 9									Within cluster sum of squared errors: 594.6409667274219		
Initial starting points (random):											
Cluster 0: 4,1,1,4,1,1											
Cluster 1: 4,2,5,10,2,1											
Cluster 2: 4,2,2,3,1,2,1											
Cluster 3: 1,2,1,2,1,1											
Cluster 4: 1,2,2,4,1,1											
Cluster 5: 1,2,2,1,8,1,1											
Cluster 6: 2,2,1,10,2,1											
Cluster 7: 0,0,1,1,1,1											
Missing values globally replaced with mean/mode											
Final cluster centroids:											
		Attribute	Full Data	0	1	2	3	4	5	6	7
			(4773.0)	(534.0)	(384.0)	(236.0)	(869.0)	(686.0)	(181.0)	(547.0)	(1336.0)
<hr/>										<hr/>	<hr/>
		Idade	1.7461	4	2.1432	4	1.0311	1.9519	1.7735	2	0.5846
		Atraso	2.4588	1.2772	0.7969	6	6	1.0015	6	1.2742	1.2335
		Valor	1.4718	1.2247	5	0.4576	0.3751	2.3601	0.6796	1.2486	1.1916
		CONTATO	4.7591	4.161	5.8203	3.9958	4.7135	5.3353	4.9503	4.5027	4.6407
		EFEITIVO	1.1672	1.1685	1.2422	1.2246	1.1749	1.1064	1.1381	1.1828	1.1587
		Acordo	0.8184	1	1	1	1	0	0	1	1

Figura 13 – Saída do algoritmo para 8 clusters

Fonte: Acervo do conteudista

Segue a figura 14 com visualização gráfica para os 8 clusters.

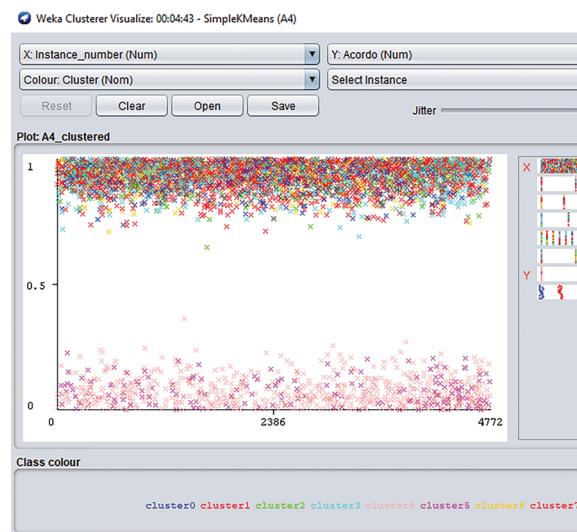


Figura 14 – Visualização gráfica para 8 clusters

Fonte: Acervo do conteudista

Ao se chegar em 11 clusters o algoritmo se comporta bem, dividindo novamente os que fecham acordo e os que não, segue a figura 15 com a visualização gráfica para 11 clusters.

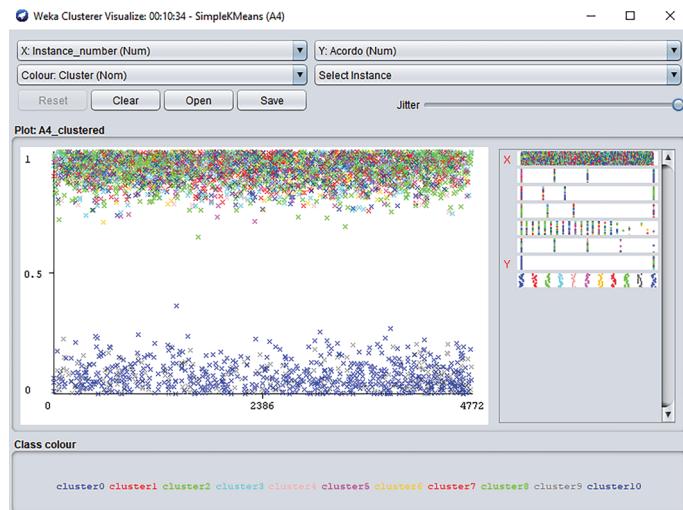


Figura 15 – Visualização gráfica para 11 clusters

Fonte: Acervo do conteudista

Note que o algoritmo cria 2 grupos que não fecham acordo e outros 9 grupos que fecham acordo, novamente uma divisão precisa dos grupos. Note que o erro RMS para esse modelo ficou em “515.1967400904064”, que já não é uma diferença tão grande se comparado ao modelo gerado para 7 clusters, ou seja, a criação de mais *clusters* daí em diante já não agrupa ao modelo, ou seria desperdício de recursos, caso a escala de experimento seja muito grande.

Todas as execuções do algoritmo de 7 *clusters* em diante não agregam mais ao modelo, hora ele diminui e hora aumenta o erro RMS, sendo assim, não sendo mais necessárias novas execuções.

Seguem os valores de erro RMS para cada um dos modelos gerados, partindo de 1 e indo até 1471 *clusters*, certamente não foram executados todos um a um, perceba através da tabela que depois de 15 *clusters* o salto foi aumentando.

Tabela 1

Cluster	Erro RMS
1	2590.5480758155823
2	1861.1424641830763
3	1293.9811248173241
4	1010.776154354433
5	909.7776150855439
6	760.3477849099173
7	712.1244016805499
8	594.6409667274219
9	674.3703620353895
10	628.5464631055665
11	515.1967400904064
12	437.79620334747824
13	427.4340398244632

<i>Cluster</i>	Erro RMS
14	422.22307302215694
15	410.06087155939133
25	286.6042647645641
40	231.33716820201462
60	158.3244558724096
100	106.10783043200178
300	40.14121292350501
500	21.974279778760614
1000	4.665173504438839
1471	0.0

A importância do experimento com o número enorme de *clusters* é para ilustrar que com o número de *clusters* tendendo ao número de instância o ERRO RMS chega em “0”, ou seja, porque cada clusters será uma própria instância e não terá erro algum.

Segue a figura 16 com o decaimento do erro RMS nos modelos gerados com *clusters* entre 1 e 15.

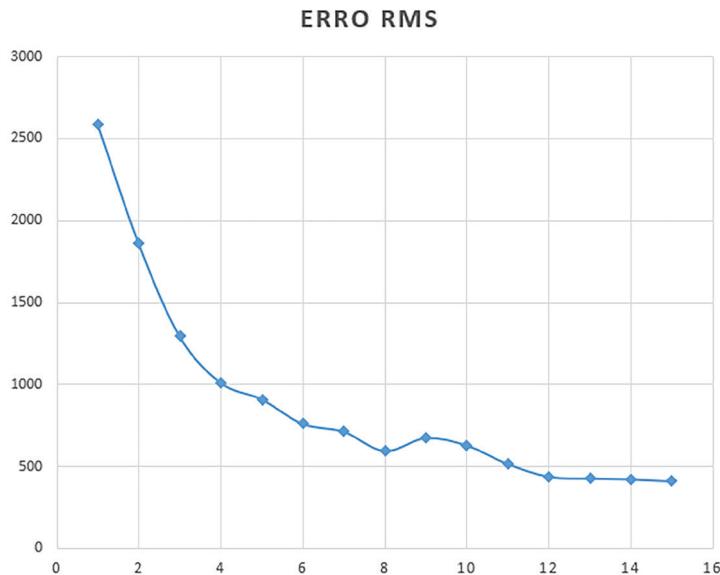


Figura 16 – Gráfico com o decaimento do ERRO RMS

Fonte: Acervo do conteudista

Observe que para 1 único *cluster* o erro RMS é bastante alto, porém à medida que aumentamos o número de *clusters* vai caindo, quando estamos com o número de *clusters* ótimo para o modelo ocorre o que chamamos de “joelho da curva”, ou seja, no momento em que o decaimento é menor, mas tenderá a “0” chegando ao número total de instâncias do modelo. Para o experimento aqui exibido a melhor quantidade de *clusters* é “7”, a partir desse número decaimento varia, e certamente não há a necessidade de continuar gerando mais *clusters*, essa seria uma condição de parada para a elaboração de um modelo a partir do algoritmo *kmeans*.

Material Complementar

Indicações para saber mais sobre os assuntos abordados nesta Unidade:

Leitura

10 Interesting Use Cases for the K-Means Algorithm

<http://bit.ly/2THV8CK>

Referências

OWEN, S.; ANIL, R.; DUNNING, T.; FRIEDMAN, E. *Mahout in Action*. Manning Publications Co., Greenwich, CT, USA, 2011.

THEODORIDIS, S.; KOUTROUMBAS, K. *Pattern Recognition*, Fourth Edition. 4th. ed. [S.l.]: Academic Press, 2008.

SOUZA, A. M. da C. **Uma nova arquitetura para Internet das Coisas com análise e reconhecimento de padrões e processamento com Big Data. 2015.** Tese (Doutorado em Sistemas Eletrônicos) - Escola Politécnica, Universidade de São Paulo, São Paulo, 2015. Disponível em: <<http://www.teses.usp.br/teses/disponiveis/3/3142/tde-20062016-105809/>>. Acesso em: 2017-03-07



Cruzeiro do Sul
Educacional