UNIVERSIDADE ESTADUAL DO OESTE DO PARANÁ
CENTRO DE CIÊNCIAS EXATAS E TECNOLÓGICAS
COLEGIADO DE CIÊNCIA DA COMPUTAÇÃO

Análise de Agrupadores
Aprendizagem de Máquina

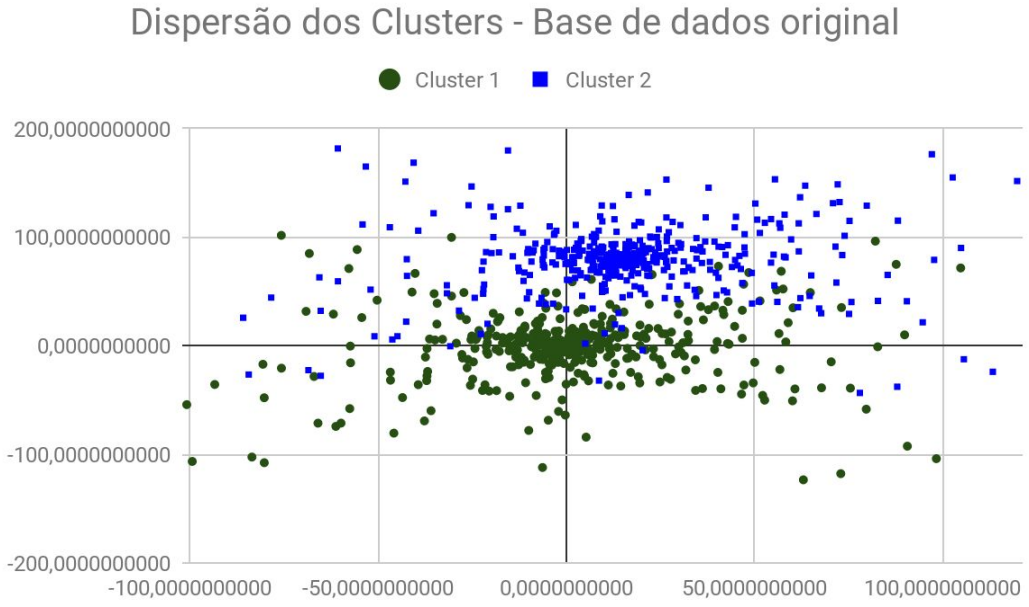
Alunos: Gilberto Antunes Monteiro Junior & Henrique Tomé Damasio
Data: 27/05/2019

Descrição do Trabalho

O objetivo do trabalho consiste em analisar o comportamento de três tipos diferentes de agrupadores, e avaliados de acordo com as abordagens de coesão, entropia, separabilidade e silhueta. Para isso, foram analisados os agrupadores implementados na biblioteca scikit learn, estes foram, DBScan, K-Means, AGNES, foi utilizada a entrada de dados com base em arquivo de formato CSV.

Descrição do Conjunto de Dados

O conjunto de dados utilizado durante os experimentos consiste em uma base composta por 1000 instâncias, as quais são divididas em apenas duas classes, 1 e 2. Dentre os 1000 exemplares do conjunto, 500 pertencem à classe 1 e 500 à classe 2. Cada instância é composta por 2 atributos, todos do tipo real.



DESCRIÇÃO PASSO-A-PASSO DO EXPERIMENTO

Inicialmente foram implementados os critérios de avaliação exigidos no trabalho, que são coesão, entropia, separabilidade e silhueta, todos foram implementados seguindo as instruções passadas em sala de aula.

Após a implementação dos critérios, foram importados os algoritmos de classificação da biblioteca scikit learn. Para aqueles que deveriam ter parâmetros otimizados, cada parâmetro foi variado dentre as seguintes séries. Para o algoritmo DBScan o parâmetro *eps* (tamanho do raio adotado) foi oscilado de 10 à 20, dentre este intervalo ruídos foram detectados, e conforme incrementado o parâmetro a quantidade de ruídos diminuiu porém sem eliminar todos. Além do parâmetro eps, ainda foi oscilado o parâmetro *min_samples* (número mínimo de pontos), este em um intervalo de 100 à 260 percorrido de 10 em 10.

Para o algoritmo K-Means os parâmetros *n_clusters* (número de centróides), *init* (posição inicial dos centróides) e *max_iter* (número de iterações para convergência) foram otimizados. *n_clusters* foi oscilado de 1 à 16, enquanto que *init* foi oscilado 2 vezes, isso ocorreu devido a forma como o algoritmo está implementado na biblioteca, para este parâmetro tem-se as seguintes opções, 'k-means++' que seleciona o centro do cluster de uma forma a acelerar a convergência e 'random' que escolhe uma posição *k* aleatória como posição inicial do centróide. Por fim o parâmetro *max_iter* foi oscilado de 50 à 650 percorrendo este intervalo de 50 em 50.

Para o algoritmo Agnes não foi necessário oscilar nenhum parâmetro. Mas a abordagem adotada pelo mesmo foi Complete Linkage (Max), portanto ao chamar o algoritmo foi passado ao mesmo o parâmetro *linkage* com a label 'complete' usando assim a distância máxima entre todos as observações entre os dois conjuntos.

Com os critérios implementados, os algoritmos importados e oscilações de parâmetro sendo realizadas bastava armazenar os resultados encontrados para cada critério de avaliação, de forma a elucidar qual estratégia e parâmetros se destacaram para cada critério, levando em consideração que todos os testes foram realizados sobre a base de dados descrita na seção anterior.

Ao executar os testes foi percebido que a estratégia DBScan apresentava um número excessivo de ruídos ao ser executada com os parâmetros descritos anteriormente, portanto para tentar minimizar esses efeitos os parâmetros foram alterados de forma que a quantidade de ruídos apresentados fosse inferior a 15%.

Para diminuir a quantidade de ruído o parâmetro *eps* foi oscilado de 26 à 33, enquanto que o parâmetro *min_samples* foi oscilado de 111 à 301. Para ambos os parâmetros o incremento era de uma unidade. Não somente os parâmetros foram alterados, mas também uma nova regra foi criada, esta implica em: **SE (possui_ruido E numero_de_ruidos < 150 E *numero_de_clusters* > 2) OU (*não_possui_ruido E numero_de_clusters > 1*)**. Esta regra garante que somente aqueles parâmetros que proporcionem um cenário com mais de um custer e com 150 ou menos ruídos serão analisados.

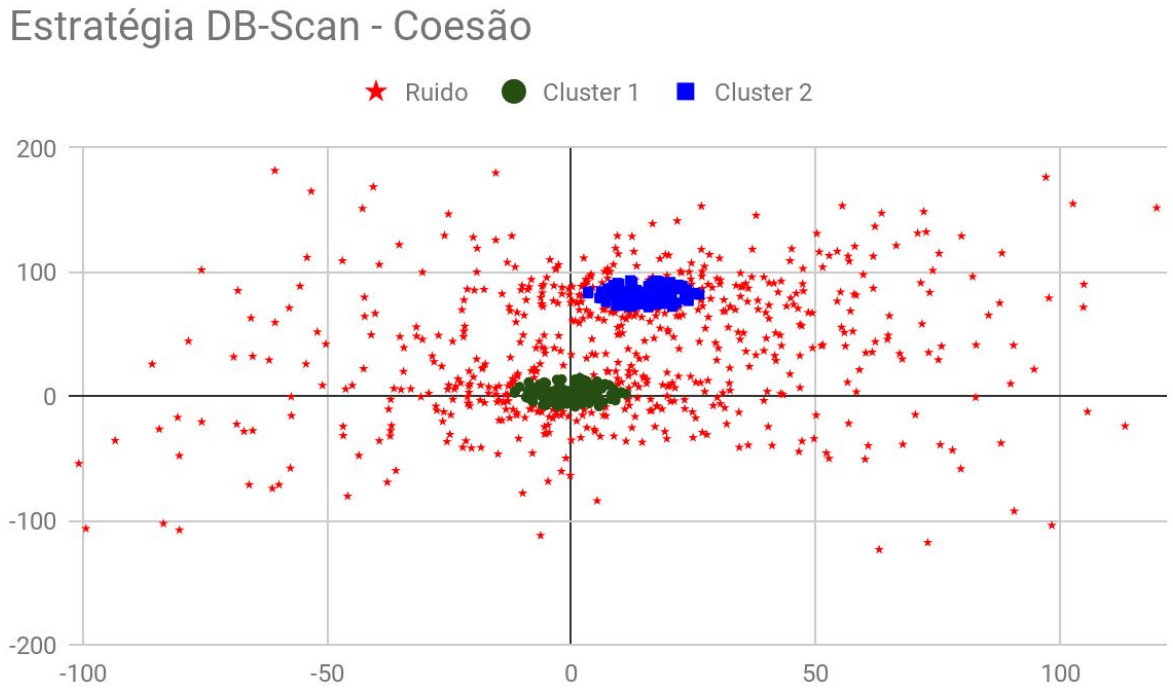
Para melhor avaliação das abordagens foi decidido fazer uma análise com ambas

configurações, portanto em um primeiro momento serão apresentados os resultados para cada abordagem segundo a primeira configuração de parâmetros (quantidade maior de ruídos), e em um segundo momento apresentaremos os resultados para cada abordagem de acordo com a segunda configuração de parâmetros (quantidade menor de ruídos). Desta forma poderemos evidenciar melhor os efeitos da escolha dos parâmetros para as estratégias.

AVALIAÇÃO DAS ABORDAGENS

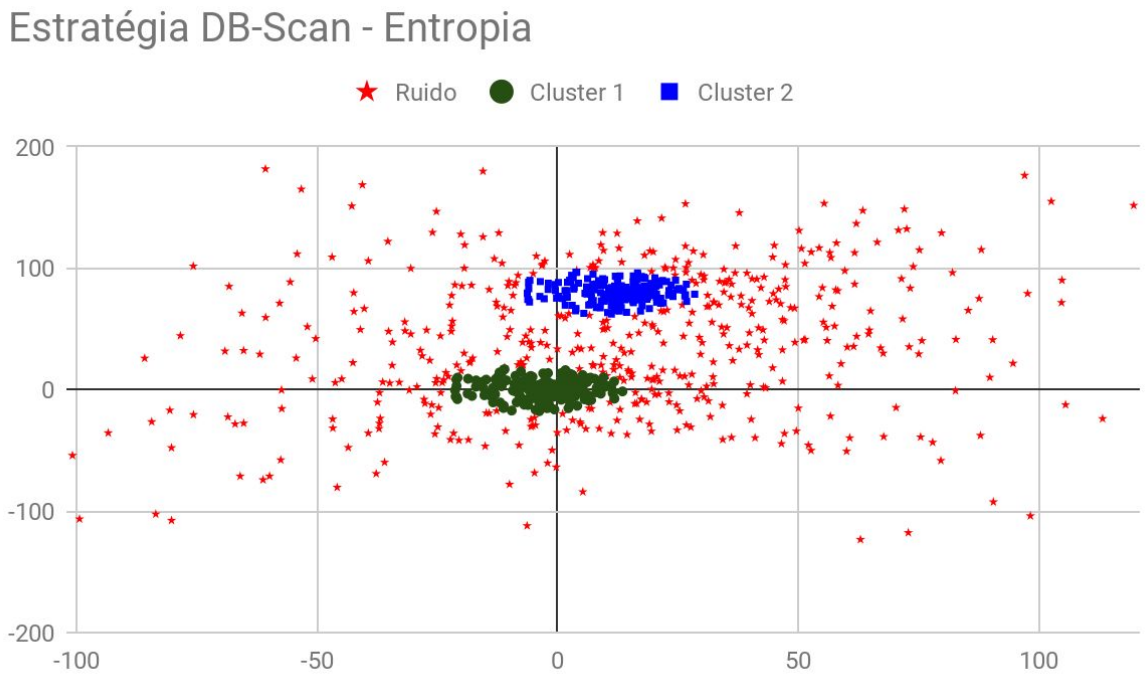
Com os algoritmos implementados, foi-se realizado o experimento descrito no item anterior, e visando obter o menor valor *SSE* (*Error Sum of Squares*) para coesão, o menor valor de entropia (próximo a zero), o maior valor *SSE* para a separabilidade e o valor o mais próximo de um para a silhueta, assim os resultados para cada abordagem e para a primeira configuração de parâmetros foram:

- Coesão: A estratégia *DB-Scan* foi a melhor, com um valor resultado de *SSE* = 10491.347027733262, utilizando os parâmetros, *eps* = 11 e *min_samples* = 190.
- Entropia: Novamente a estratégia DB-Scan foi a melhor, com um valor resultado de entropia = 0.035925580783167384, utilizando os parâmetros, *eps* = 18 e *min_samples* = 260.
- Separabilidade: A estratégia *K-Means* foi a melhor, com um valor resultado de *SSE* = 15003.973801811582, utilizando os parâmetros, *n_clusters* = 9, *init* = '*random*' e *max_iter* = 200.
- Silhueta: E mais uma vez a estratégia *DB-Scan* foi a melhor, com um valor resultado de entropia = 0.9031784358523376, utilizando os parâmetros, *eps* = 11 e *min_samples* = 190.

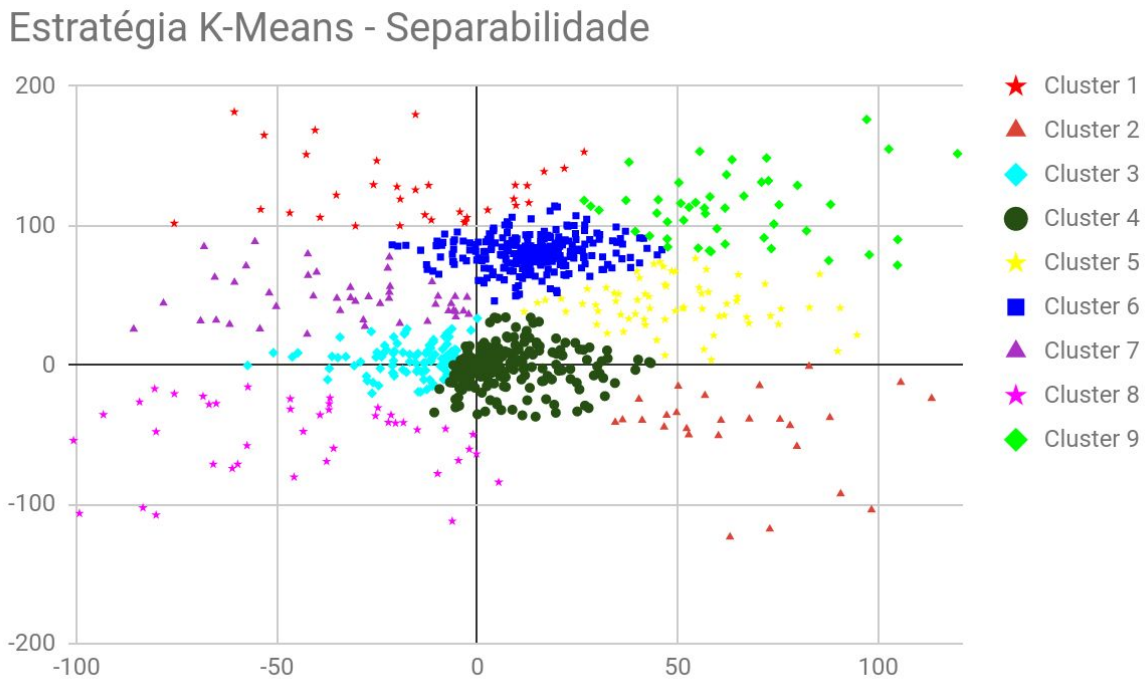


Devido a estratégia DBScan trabalhar com agrupamento por densidade grande parte dos dados acabam sendo considerados como ruído e isso acarreta em um baixo valor de coesão

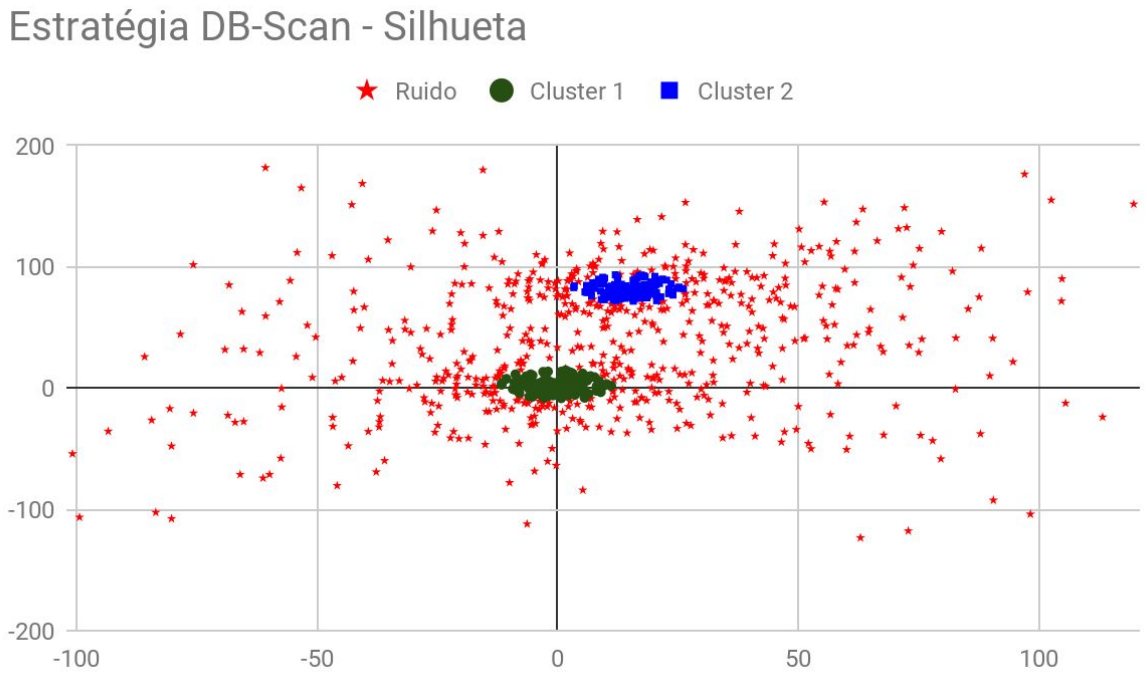
pelo fato de tanto o cluster 1 quanto o cluster 2 possuírem elementos bastante próximos uns dos outros. Neste cenário seria mais interessante ter um valor de coesão maior visando a diminuição da quantidade de ruídos. É importante ressaltar que o gráfico acima foi construído utilizando como base o agrupamento do algoritmo DBScan tendo como parâmetros *eps* = 11 e *min_samples* = 190.



O cálculo da entropia tem por finalidade medir a desordem em um sistema, e comparando o gráfico da seção 2, o qual demonstra a dispersão dos dados na base original, com este gráfico podemos notar que o método DBScan acabou por separar os clusters 1 e 2 de forma que estes ficassem o menos desordenados possível, alocando elementos pertencentes ao cluster 1 (original) para o cluster 1 (após o agrupamento) e alocando os elementos pertencentes ao cluster 2 (original) para o cluster 2 (após o agrupamento). Os parâmetros utilizados para gerar este gráfico foram, *eps* = 18 e *min_samples* = 260.



Pelo fato da estratégia K-Means realizar um agrupamento por particionamento podemos notar que aqui não existem ruídos, ou seja, todos os elementos foram alocados em algum cluster, assim sendo a distância média de um cluster em relação aos outros acaba sendo maior se comparado com uma estratégia que detecta ruídos.



Silhueta refere-se a um método de interpretação e validação de consistência dentro de clusters

de dados . A técnica fornece uma representação de quão bem cada objeto foi classificado. Novamente tendo como base o gráfico da base de dados original podemos validar essa definição pela comparação dos gráficos. Apenas observando os gráficos fica nítido que o algoritmo DBScan agrupou elementos do cluster 2 (original) no cluster 1 (após o agrupamento) e vice versa. Mas também pode-se observar que isso não ocorre com muitos elementos, portanto se justifica o valor de 0,903178 apresentado já que os objetos foram muito bem classificados. Vale ressaltar que os parâmetros utilizados foram *eps* = 11 e *min_samples* = 190.

Os resultados para todas as estratégias segundo cada abordagem pode ser vista na Tabela 1.

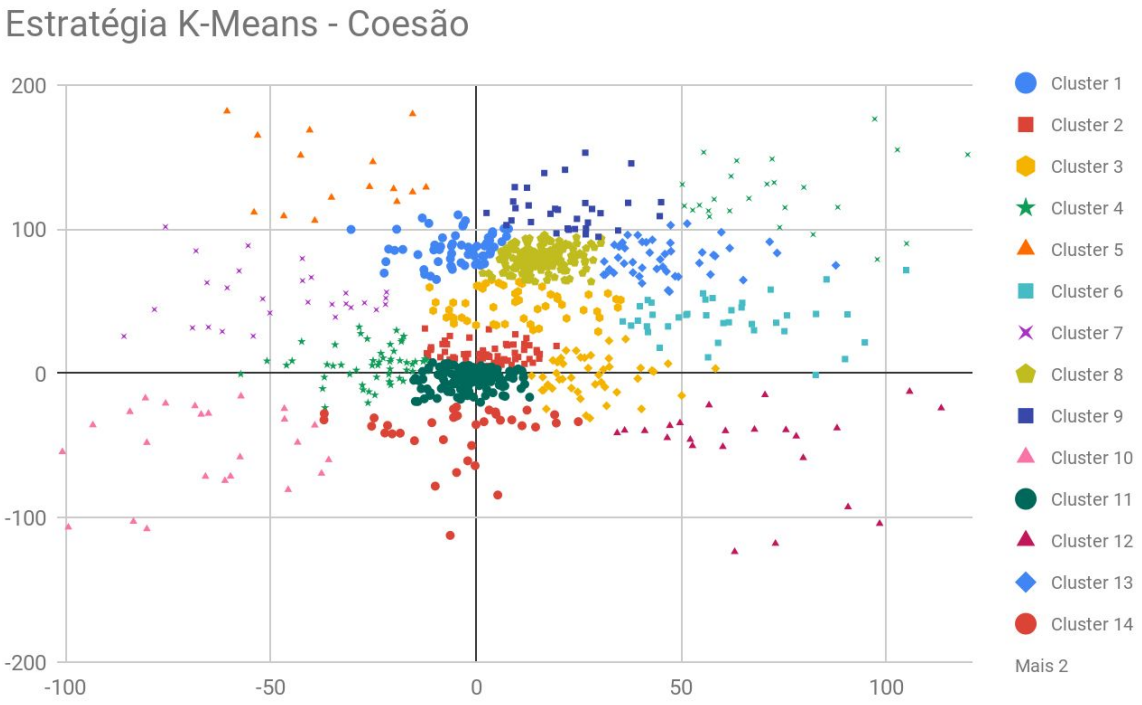
Tabela 1 : Resultados das estratégias entre as diferentes abordagens

	Coesão	Entropia	Separabilidade	Silhueta
Agnes	1395206,214	0,3988869303	10375,3422	0,5655356026
Parâmetros	*linkage = 'complete'*	*linkage = 'complete'*	*linkage = 'complete'*	*linkage = 'complete'*
DBScan	10491,34703	0,03592558078	7116,776081	0,9031784359
Parâmetros	*eps* = 11, *min_samples* = 190	*eps* = 18, *min_samples* = 260	*eps* = 20, *min_samples* = 100	*eps* = 11, *min_samples* = 190
DBScan	879390,38748	0,28658748619	7545,9636092	0,6880119601
Parâmetros	*eps* = 30, *min_samples* = 270	*eps* = 26, *min_samples* = 116	*eps* = 32, *min_samples* = 281	*eps* = 26, *min_samples* = 116
K-Means	32142,29151	0,3544900055	15003,9738	0,6012801695
Parâmetros	*n_clusters* = 16, *init* = *'random'*, *max_iter* = 200	*n_clusters* = 2 *init* = *'random'*, *max_iter* = 50	*n_clusters* = 9, *init* = *'random'*, *max_iter* = 200	*n_clusters* = 7, *init* = *'random'*, *max_iter* = 100

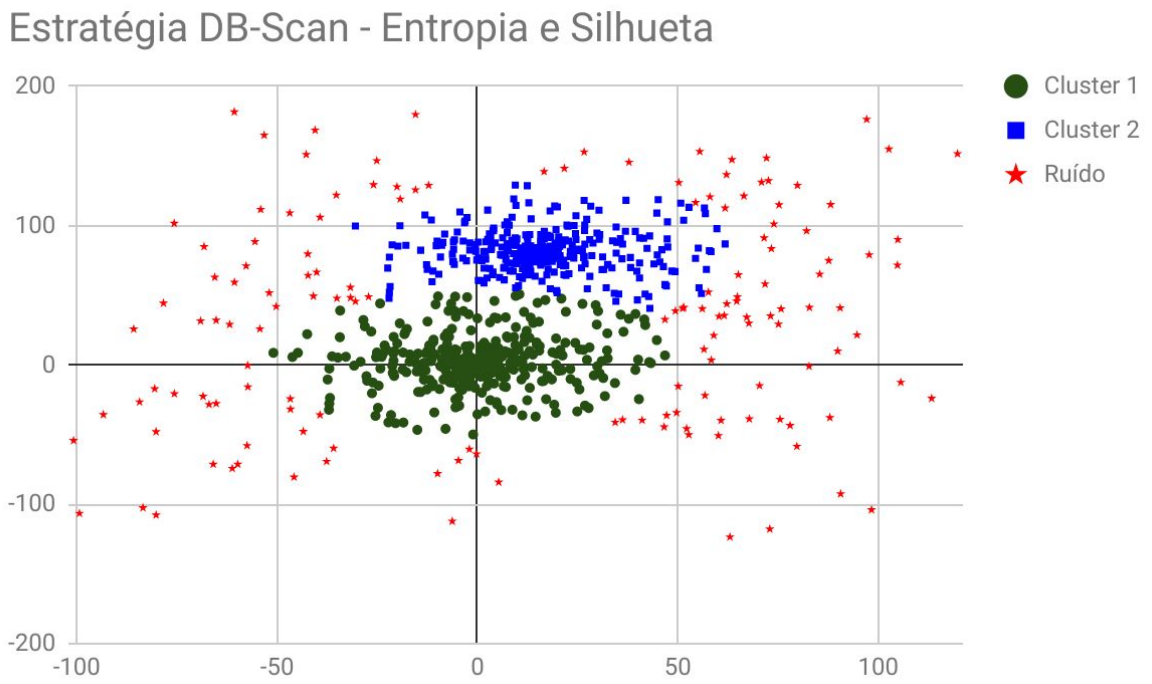
Foi possível observar que o método DBScan se destacou em relação ao K-Means e ao Agnes, mas como dito na seção anterior e no início desta seção toda a análise realizada até aqui foi baseada na primeira configuração de parâmetros (quantidade maior de ruídos). Com esta configuração os ruídos chegaram a representar até 60% das instâncias, como já discutido na seção anterior, por esse número de ruídos ser extremamente alto foi realizada uma segunda configuração de parâmetros visando manter a quantidade de ruídos menor que 10% das instâncias. Os resultados obtidos para a execução para cada abordagem de acordo com a segunda configuração foram:

- Coesão: A estratégia *K-Means* foi a melhor, com um valor resultado de *SSE* = 32142,29151, utilizando os parâmetros, *n_cluster* = 16, *init* = 'random' e *max_iter* = 200.

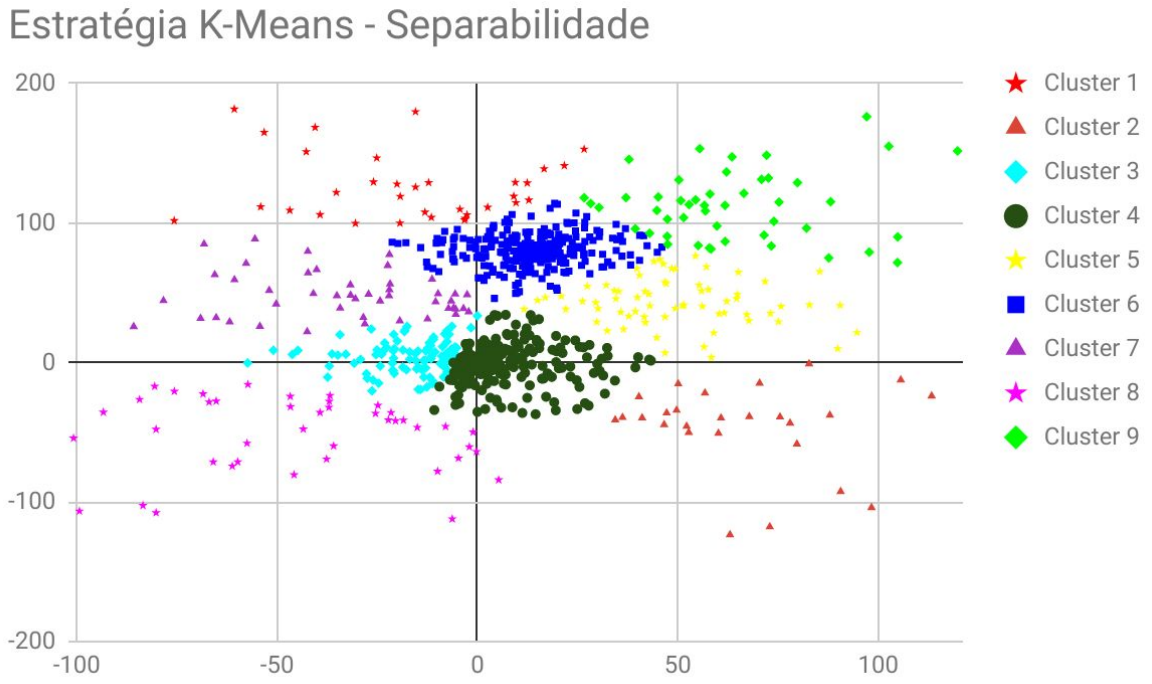
- Entropia: A estratégia *DB-Scan* foi a melhor, com um valor resultado de entropia = 0.2865874861864468, utilizando os parâmetros, *eps* = 26 e *min_samples* = 116.
- Separabilidade: A estratégia *K-Means* foi a melhor, com um valor resultado de *SSE* = 15003,9738, utilizando os parâmetros, *n_cluster* = 9, *init* = 'random' e *max_iter* = 200.
- Silhueta: E mais uma vez a estratégia *DB-Scan* foi a melhor, com um valor resultado de entropia = 0.6880119601377823, utilizando os parâmetros, *eps* = 26 e *min_samples* = 116.



Com a segunda configuração de parâmetros a estratégia K-Means acabou por apresentar um valor médio de coesão menor, e podemos entender o motivo visto que a base acabou sendo dividida em 16 clusters com um número de instâncias menores, fazendo assim clusters mais coesos.



Tanto a entropia como a silhueta apresentaram as mesmas configurações de parâmetros para a estratégia DB-Scan, podemos notar que as instâncias apresentam um certo grau de desordem, e ao mesmo tempo, ao comparar este gráfico ao da seção 2 pode-se perceber que houve certa inconsistência no agrupamento já que o cluster 1 (depois do agrupamento) possui instâncias do cluster 2 (original) e vice versa.



Curiosamente a separabilidade se manteve inalterada, isso pode ser explicado pelo fato de que mesmo após a redução dos ruídos os mesmo não foram totalmente removidos. E como todas as instâncias foram alocadas para algum cluster na estratégia K-Means a distância média de um cluster em relação aos outros acaba sendo maior.