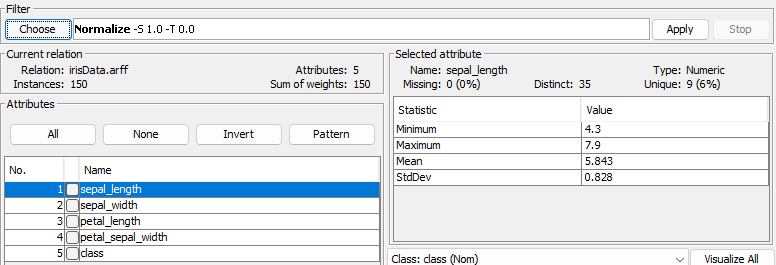
*Resolução das questões da prática 2*

* ***Prints Referentes à questão C***

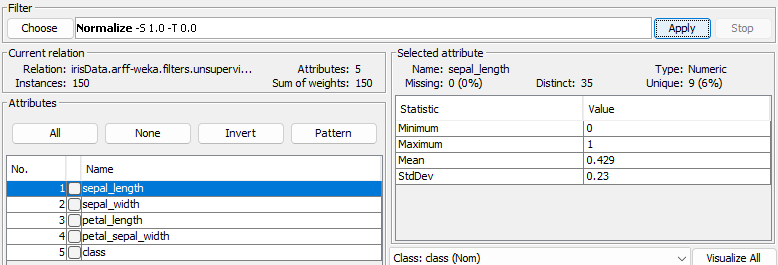
Aplicação de filtros à **iris.arff** database:

1. Normalização

Sem o filtro:



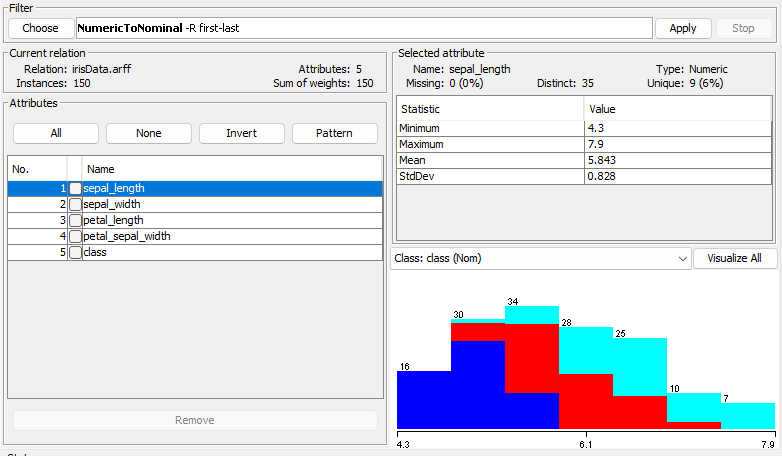
Com o filtro:



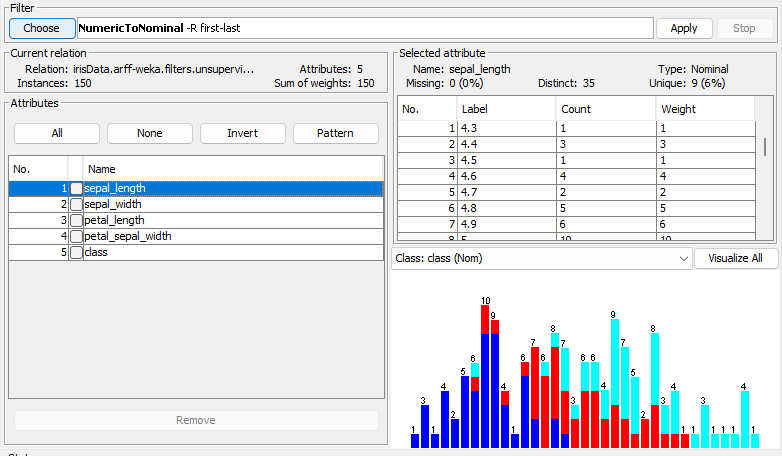
A aplicação do filtro de normalização ajusta os valores de dados para um intervalo entre 0 e 1, com o objetivo de que todos os valores da base estejam na mesma escala independentemente do valor original. Como é possível observar pelos prints, os valores mínimo e máximo foram convertidos para 0 e 1 respectivamente (gerando variações no desvio padrão e na média dos dados).

1. Numeric to Nominal

Sem o filtro:



Com o filtro:



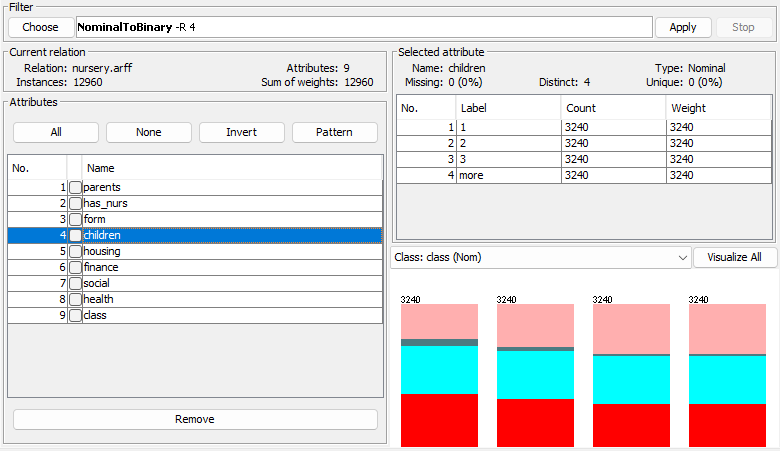
A aplicação do filtro *NumericToNominal* foi utilizado para transformar atributos numéricos em nominais, ou seja, os valores passam a descrever qualidades sem implicar em uma ordem hierárquica. O filtro pega todos os valores numéricos e os adiciona à lista de valores nominais desse atributo.

Vale ressaltar que neste caso, por mais que a transformação seja possível, não faria muito sentido que os dados sejam nominais, uma vez que os valores do dataset não são qualitativos, mas sim quantitativos.

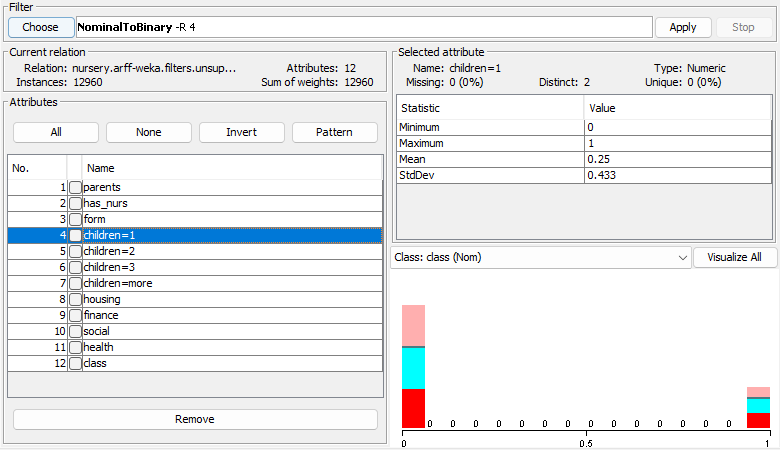
Aplicação de filtros à **nursery.arff** database:

1. NominalToBinary

Sem o filtro:



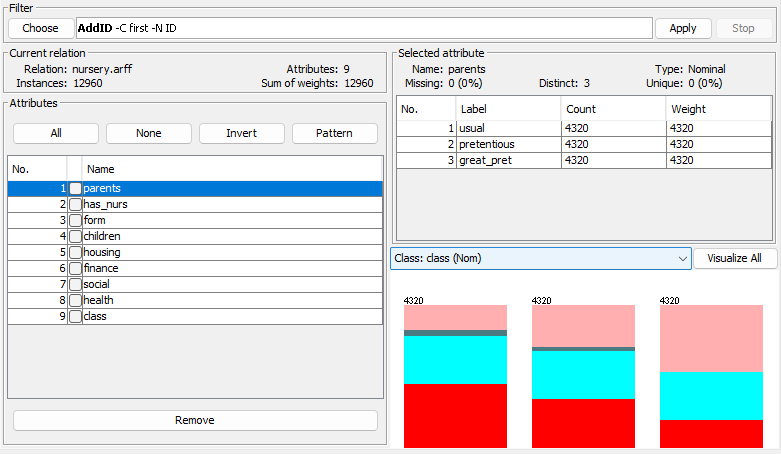
Com o filtro:



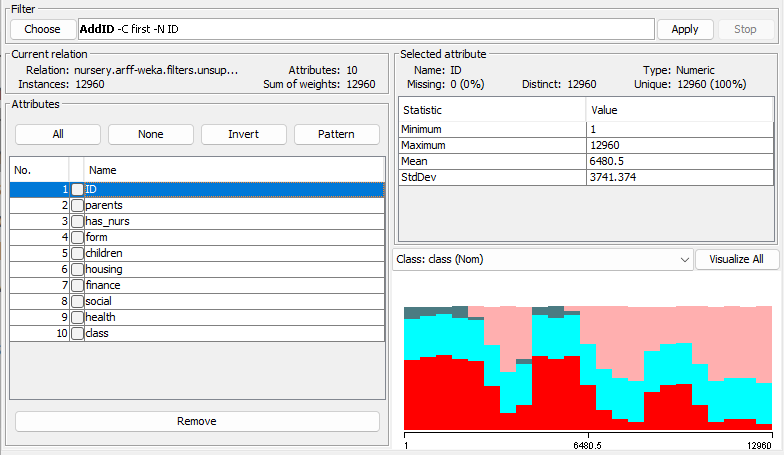
A aplicação do filtro *NominalToBinary* tem como objetivo converter todos os atributos nominais em atributos numéricos binários. A operação do filtro segue a seguinte lógica: *Um atributo com k valores é transformado em k atributos binários se a classe for nominal*. Com isso, os 4 valores nominais presentes no atributo ‘children’ serão transformados em 4 novos atributos binários (children=1, children=2, children=3, children=more)

1. AddID

Sem o filtro:



Com o filtro:

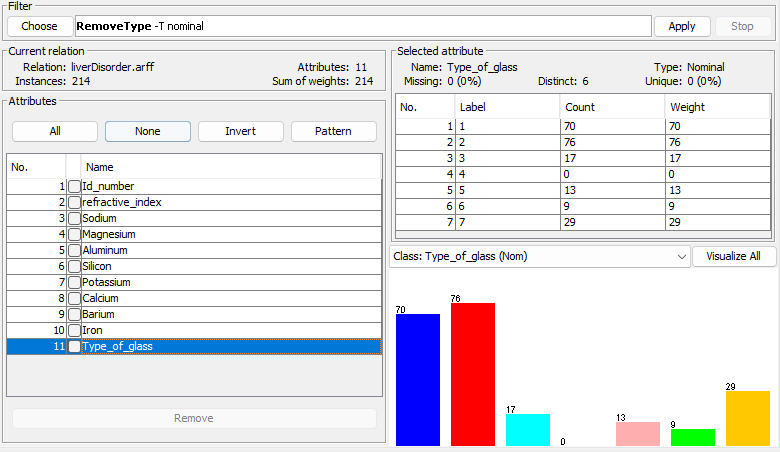


A aplicação do filtro *AddID* tem como objetivo adicionar um valor de identificação para cada um dos objetos, criando um novo atributo nomeado como ‘ID’, contendo os valores sequenciais de 1 até N (sendo N, a quantidade de instâncias no dataset). No caso da aplicação acima, o atributo contém valores numerados de 1 a 12960.

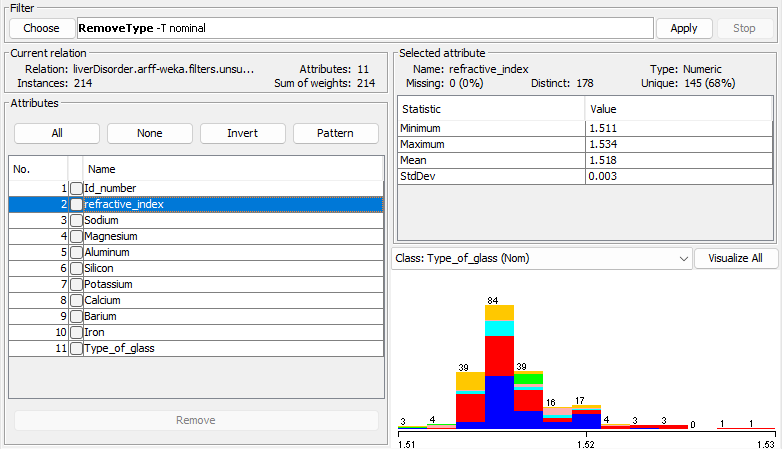
Aplicação de filtros à **glass.arff** database:

1. Remove Type (nominal)

Sem o filtro:



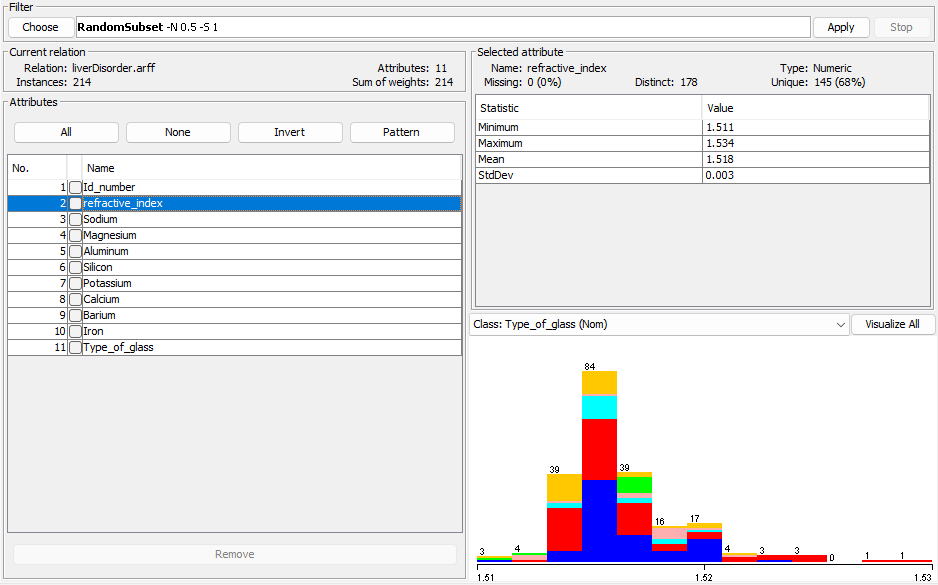
Com o filtro:



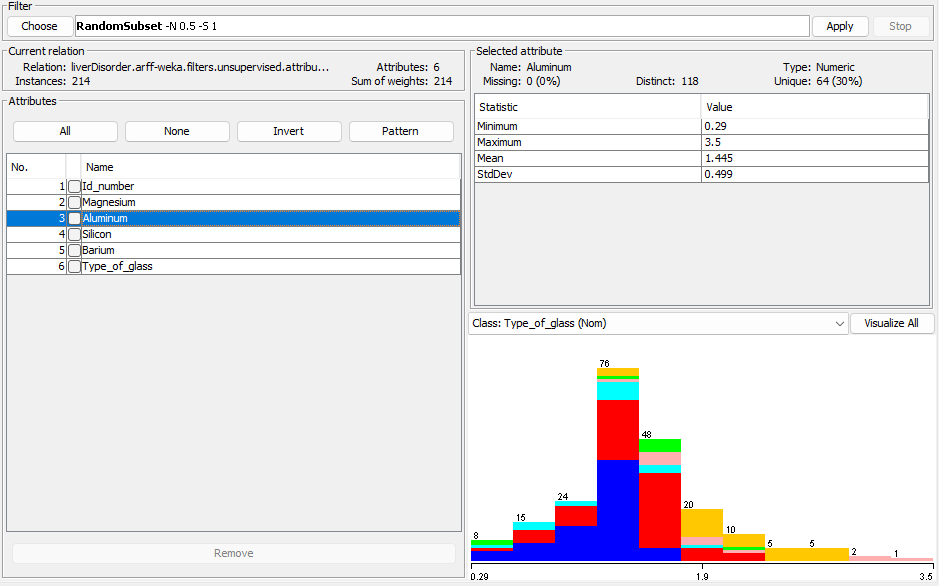
A aplicação do filtro *RemoveType* tem como objetivo retirar da base de dados todos os objetos de algum determinado tipo. Neste caso foram retirados os dados nominais da base. No caso desta aplicação o atributo de classe (Type\_of\_glass) foi excluído inteiramente, visto que sua composição são valores nominais que descrevem o tipo de vidro da base.

1. Random Subset (N 0.5, - S 1)

Sem o filtro:



Com o filtro:



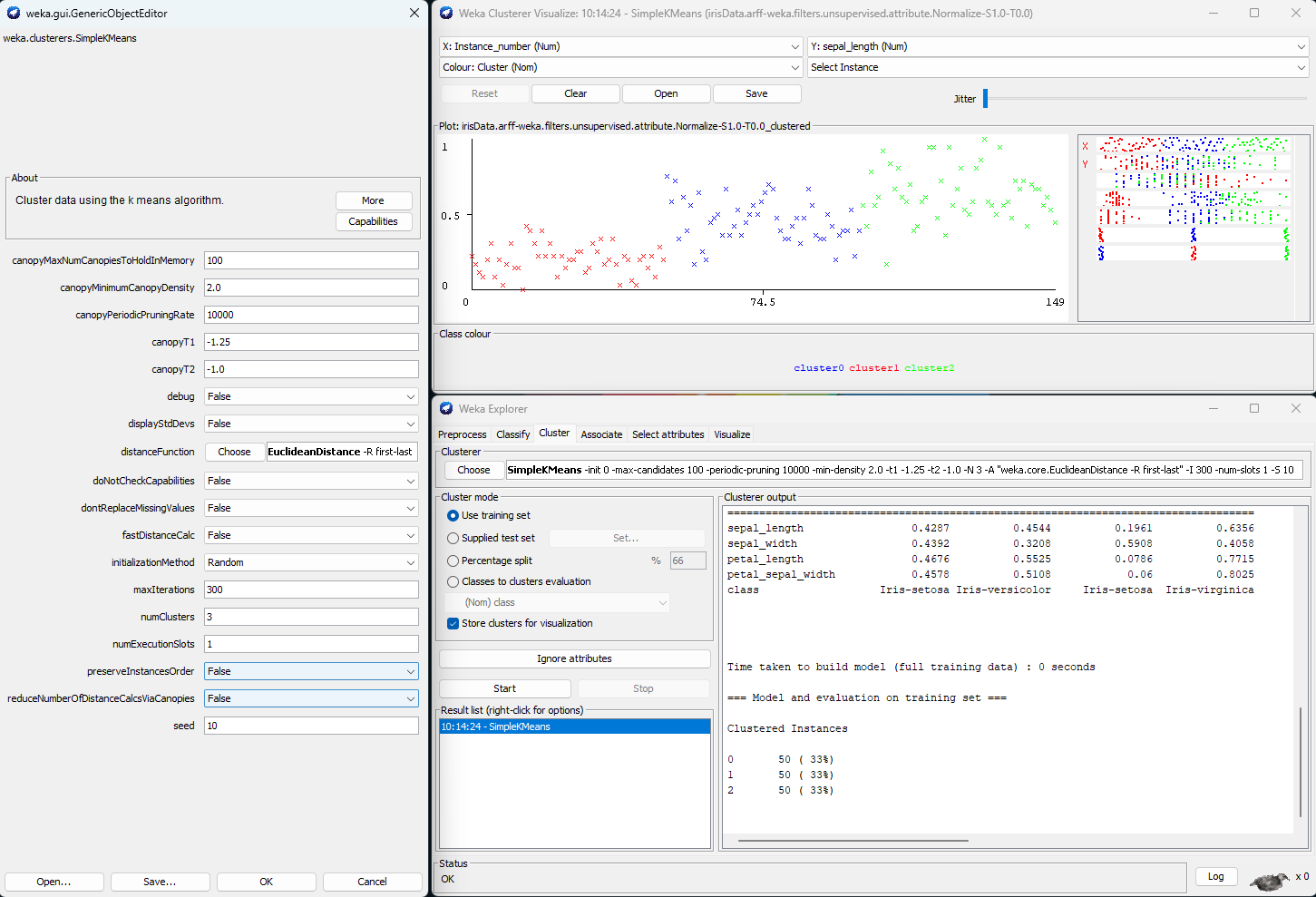
A aplicação do filtro *RandomSubset* retorna um sub-grupo de dados a partir do dataset original. De acordo com os parâmetros (N 0.5), serão selecionados 50% dos atributos originais, com isso o retorno do filtro são apenas 6 dos 11 atributos iniciais.

* ***Prints Referentes à questão D***

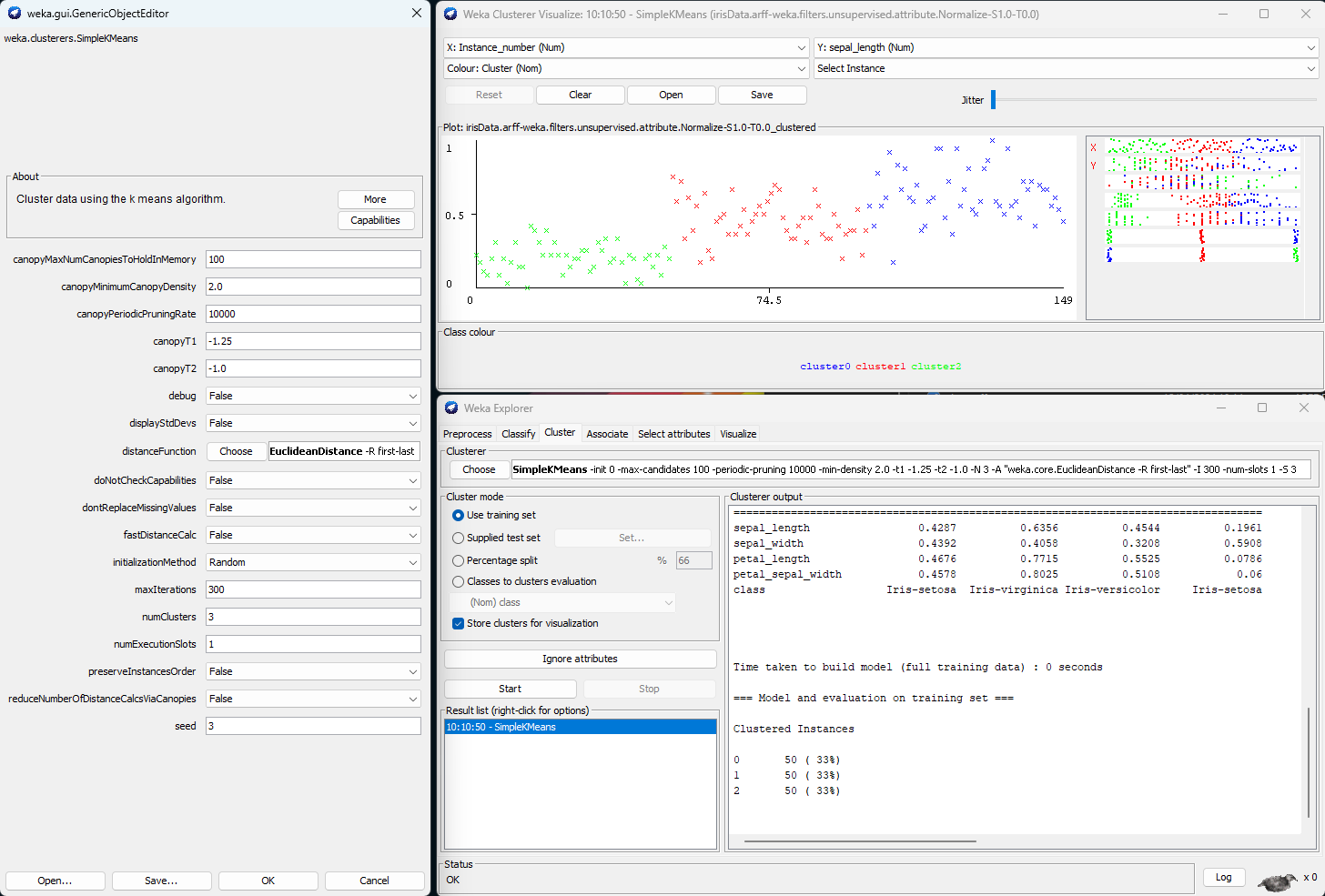
K-means aplicado no dataset ***íris***:

**obs:** Foi aplicado o filtro de normalização à base de dados originais, além disso, o atributo de classe foi removido

1. Semente geradora = 10



1. Semente geradora = 3

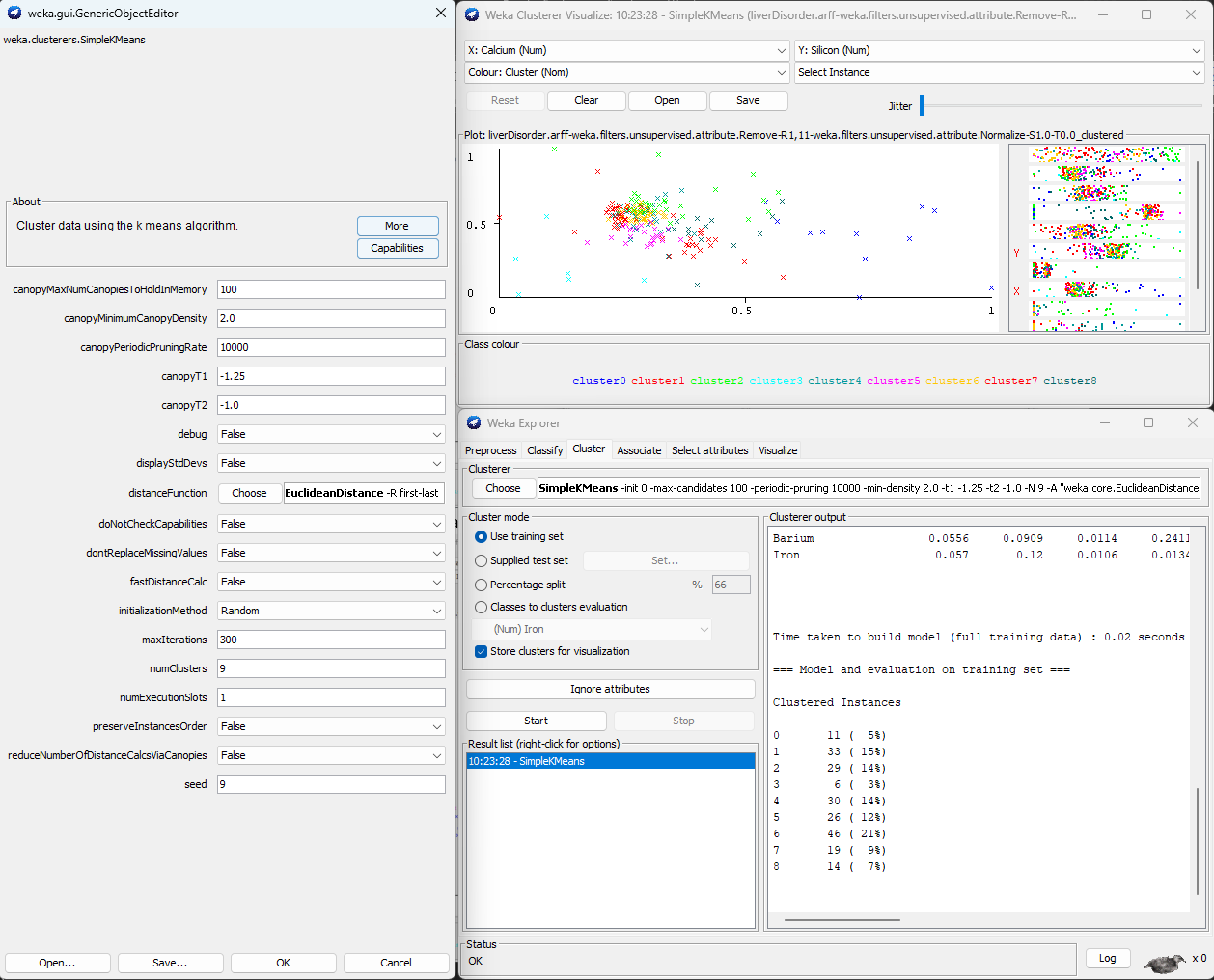


As diferentes sementes geradoras utilizadas não representam nenhuma alteração no *dataset* íris. Em ambas as aplicações os 3 clusters contém 50 elementos (representando 33% do dataset) e em uma análise visual, os clusters contém a mesma distribuição de elementos. Quando o resultado do agrupamento se mantém o mesmo, independentemente dos diferentes valores para a semente geradora no algoritmo k-means, isso geralmente indica que os clusters encontrados são relativamente robustos e estáveis.

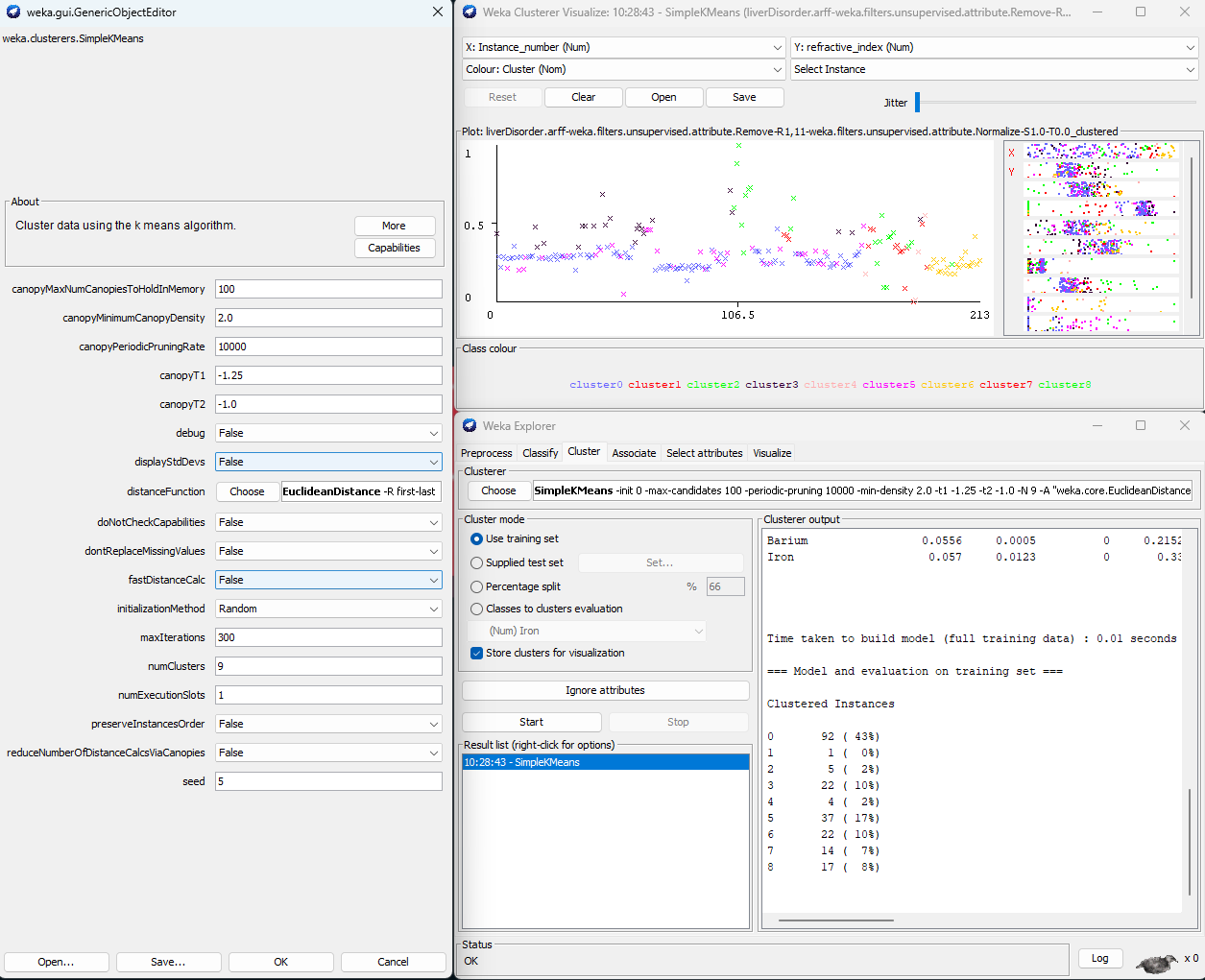
K-means aplicado no dataset ***glass***:

**obs:** Foi aplicado o filtro de normalização à base de dados originais, além disso, o atributo de classe foi removido

1. Semente geradora = 9



1. Semente geradora = 5



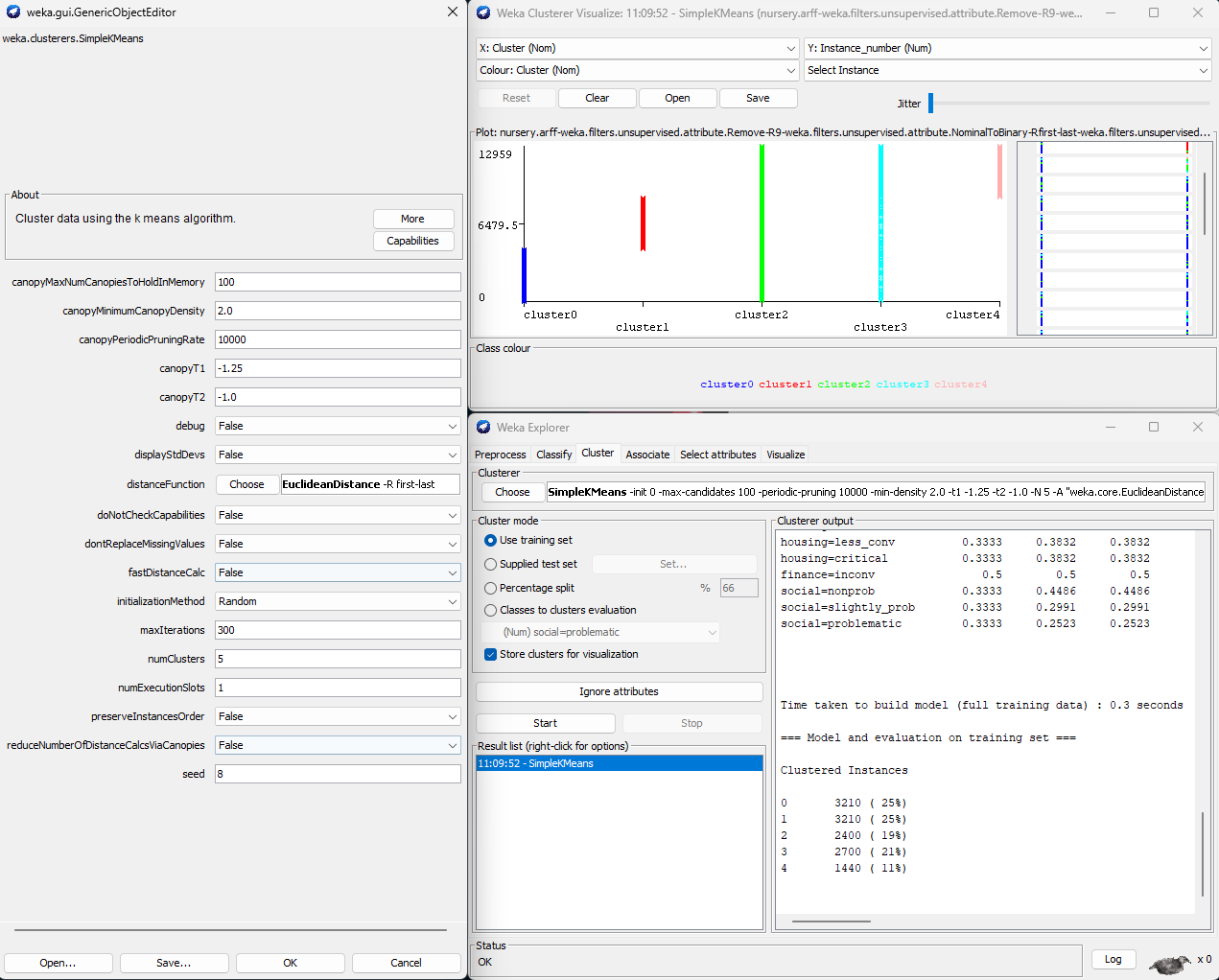
A aplicação dos k-means resultou em diferentes resultados de agrupamento com as diferentes sementes geradoras, é importante ressaltar que uma mesma semente geradora sempre resulta nos mesmos resultados de agrupamento. A semente geradora é utilizada para permitir que um algoritmo não-determinístico como o k-mean se comporte de maneira determinística (gera resultados fixos).

Neste caso, os diferentes resultados apresentados pelos diferentes valores da semente geradora podem indicar que os dados não estão distribuídos de forma estável, também pode indicar outros problemas como uma má separabilidade dos cluster ou uma definição inadequada do número ideal de clusters

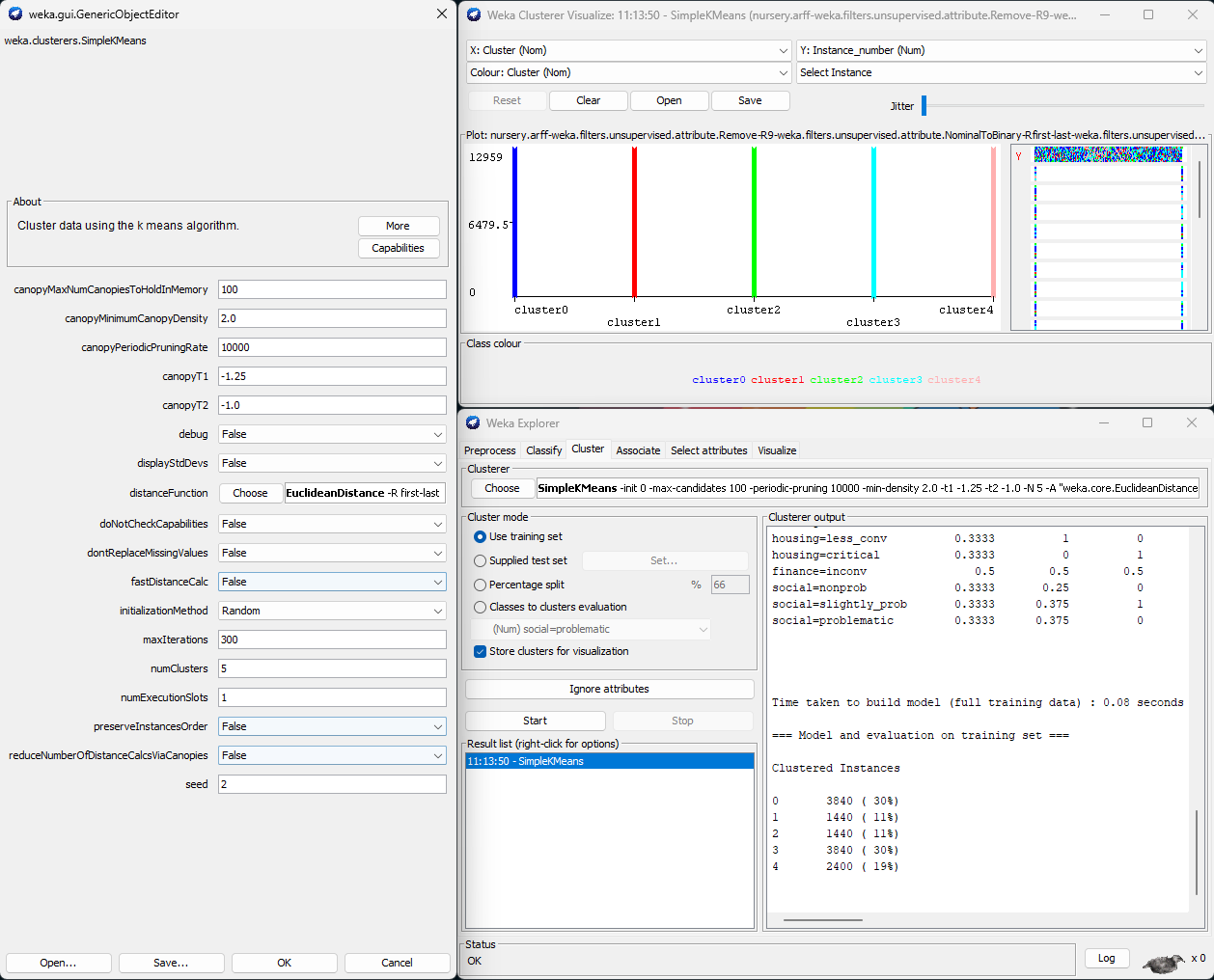
Kmeans aplicado no dataset ***nursery***:

**obs:** O dataset *nursery* contém apenas dados Nominais, para que fosse possível aplicar o algoritmo de agrupamento k-means foi necessário realizar uma transformação nos dados, de modo que cada atributo nominal fosse convertido em valores binários

1. Semente geradora = 8



1. Semente geradora = 2



A aplicação dos k-means resultou em diferentes resultados de agrupamento com as diferentes sementes geradoras, da mesma forma que ocorre durante a aplicação no dataset *glass.* É importante ressaltar que o dataset *nursery* contém valores binários e com isso, algumas questões precisam ser consideradas:

* + - * O algoritmo k-means utiliza a distância euclidiana para calcular a similaridade entre pontos. Com dados binários, onde os valores são apenas 0 ou 1, a distância euclidiana pode não ser a métrica mais apropriada, pois não leva em conta a natureza binária dos dados.
      * A interpretação dos clusters formados pode ser mais direta com dados binários, especialmente se cada dimensão binária corresponder a uma característica específica.