**Tarefa 3: conjunto de dados real**

Investiguemos agora o processo de classificação de um conjunto de dados real, o conjunto p*ima indian diabetes*. Pode ver uma descrição do conjunto de dados no repositório da UCI.

Ao treinar uma SVM é necessário definir os valores adequados para uma série de parâmetros, o parâmetro de penalização C e do *kernel*, caso seja aplicável. Nesta tarefa veremos como fazer a determinação dos referidos parâmetros.

Dividamos primeiro o conjunto de dados em duas partes, conjuntos de treino e de teste.

1. Para tal, no Explorador basta fazer o seguinte:
   1. Conjunto de treino:
      1. Carregue o conjunto de dados *diabetes*.
      2. Seleccione o filtro *RemovePercentage* no painel de pré-processamento (*filters.unsupervised.instance.RemovePercentage*).
      3. Defina a percentagem correcta para a divisão dos dados, 60 % neste caso. Para tal faça click no espaço diante do botão *Choose* para abrir as opções do filtro.
      4. Aplique o filtro.
      5. Guarde os dados gerados como um novo ficheiro (*diabetes\_trn*)
   2. Conjunto de teste:
      1. Recarregue o conjunto de dados completo (ou use apenas o botão *Undo* para reverter as alterações para o conjunto de dados).
      2. Selecione o filtro *RemovePercentage* se ainda não estiver selecionado.
      3. Defina a propriedade como *invertSelection* como verdadeira. Para tal faça click no espaço diante do botão *Choose* para abrir as opções do filtro.
      4. Aplique o filtro.
      5. Guarde os dados gerados como novo ficheiro (*diabetes\_tst*).

Determinemos o melhor conjunto de parâmetros com base nos resultados de uma validação cruzada em 5 folhas realizada sobre o conjunto de treino.

1. Carregue o conjunto de dados de treino (*diabetes\_trn*).
2. Efectue de forma sucessiva a validação cruzada para as seguintes combinações de parâmetros:
   1. Kernel RBF: C = 0,01; 0,1; 1; 10; 100 G = 0,001; 0,01; 0,1; 1,0;10
   2. Kernel polinomial: C = 0,01; 0,1; 1; 10; 100; p = 2, 4, 8, 16
   3. Kernel linear: C = 0,01; 0,1; 1; 10; 100

KernelRBF

|  |  |  |
| --- | --- | --- |
| C | Gamma | Taxa de Erro |
| 0,01 | 0,001 | 0.3029 |
| 0,1 | 0,01 | 0.0297 |
| 1 | 0,1 | 0.3029 |
| 10 | 1.0 | 0.2345 |
| 100 | 10 | 0.2943 |

Kernel Polinomial

|  |  |  |
| --- | --- | --- |
| C | P | Taxa de Erro |
| 0,01 | 2 | 0.3029 |
| 0.1 | 4 | 0.2248 |
| 1 | 8 | 0.2997 |
| 10 | 16 | 0.3094 |
| 100 |  | 0.2184 |

Kernel Linear

|  |  |
| --- | --- |
| **C** | **Taxa de Erro** |
| 0.001 | 0.3029 |
| 0.01 | 0.3029 |
| 0.1 | 0.3029 |
| 1 | 0.3029 |
| 10 | 0.3029 |
| 100 | 0.3029 |

Os menores valores da taxa de erro são registados no caso Kernel RBF quando C=0.1 e G=0.01 e no Kernel Polinomial quando C=100 e G= 1.0 (Valor por defeito) e no caso do Kernel Linear a taxa de erro é constante com o valor de 0.3029

Treinemos agora o classificador utilizando as melhores combinações dos parâmetros correspondentes a cada *kernel* e apliquemos o mesmo aos dados de teste.

Carregue o conjunto de treino.

Seleccione o classificador e defina os valores dos parâmetros a utilizar.

Seleccione *Supplied test set* nas opções de teste e carregue o ficheiro de teste.

Treine o classificador e anote os resultados da classificação para cada alternativa testada.

KernelRBF

|  |  |  |
| --- | --- | --- |
| C | Gamma | Taxa de Erro |
| 0,01 | 0,001 | 0.3796 |
| 0,1 | 0,01 | 0.3796 |
| 1 | 0,1 | 0.4228 |
| 10 | 1.0 | 0.2882 |
| 100 | 10 | 0.0087 |

.

Kernel Polinomial

|  |  |  |
| --- | --- | --- |
| C | P | Taxa de Erro |
| 0,01 | 2 | 0.3818 |
| 0.1 | 4 | 0.2364 |
| 1 | 8 | 0.1887 |
| 10 | 16 | 0. |
| 100 |  | 0.2184 |

Kernel Linear

|  |  |
| --- | --- |
| **C** | **Taxa de Erro** |
| 0.001 | 0.3796 |
| 0.01 | 0.3796 |
| 0.1 | 0.3796 |
| 1 | 0.3796 |
| 10 | 0.3796 |
| 100 | 0.3796 |

Os resultados obtidos da classificação são distintos a quando da realização da mesma com validação cruzada. Em alguns casos os valores são maiores e outros menores comparando com o caso anterior.