Tutorial for the implemented penalized approach for anomaly detection

AMVA4NewPhysics authors

2017-12-22

In the tutorial, we present how to use package PenalizedAD that contains the implementation of penalized model-based clustering approach and as well the following anomaly detection method.

First, let us load an example of the background data of size $n = 500$ and dimension $P = 16$ generated by a mixture of 2 Gaussian components with equal proportions. The Gaussian comonet means are equal respectively to $\boldsymbol{\mu}_1 = (2, ..., 2, 0, ..., 0)$ and $\boldsymbol{\mu}_2 = (-2, ..., -2, 0, ..., 0)$ where number of both zero and non-zero elements is equal to 8. Covariance matrices are set to be respectively

$$\Sigma_k = \begin{pmatrix} \Sigma_{k1} & 0 & 0 \\ 0 & \Sigma_{k1} & 0 \\ 0 & 0 & I_8 \end{pmatrix}$$

where $\Sigma_{k1} = P_k D_k P_k'$ with

$$P_1 = \begin{pmatrix} 1 & 0 & -1 & 1 \\ 1 & \sqrt{2} & 1 & 0 \\ 1 & -\sqrt{2} & 1 & 0 \\ -1 & 0 & 1 & 2 \end{pmatrix}, \quad P_2 = \begin{pmatrix} -1 & 0 & 1 & 1 \\ 1 & -\sqrt{2} & 1 & 0 \\ 1 & \sqrt{2} & 1 & 0 \\ 1 & 0 & -1 & 2 \end{pmatrix}$$

$$D_1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0.5 & 0 & 0 \\ 0 & 0 & 0.15 & 0 \\ 0 & 0 & 0 & 0.12 \end{pmatrix}, \quad D_2 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0.1 & 0 & 0 \\ 0 & 0 & 0.4 & 0 \\ 0 & 0 & 0 & 0.1 \end{pmatrix}.$$

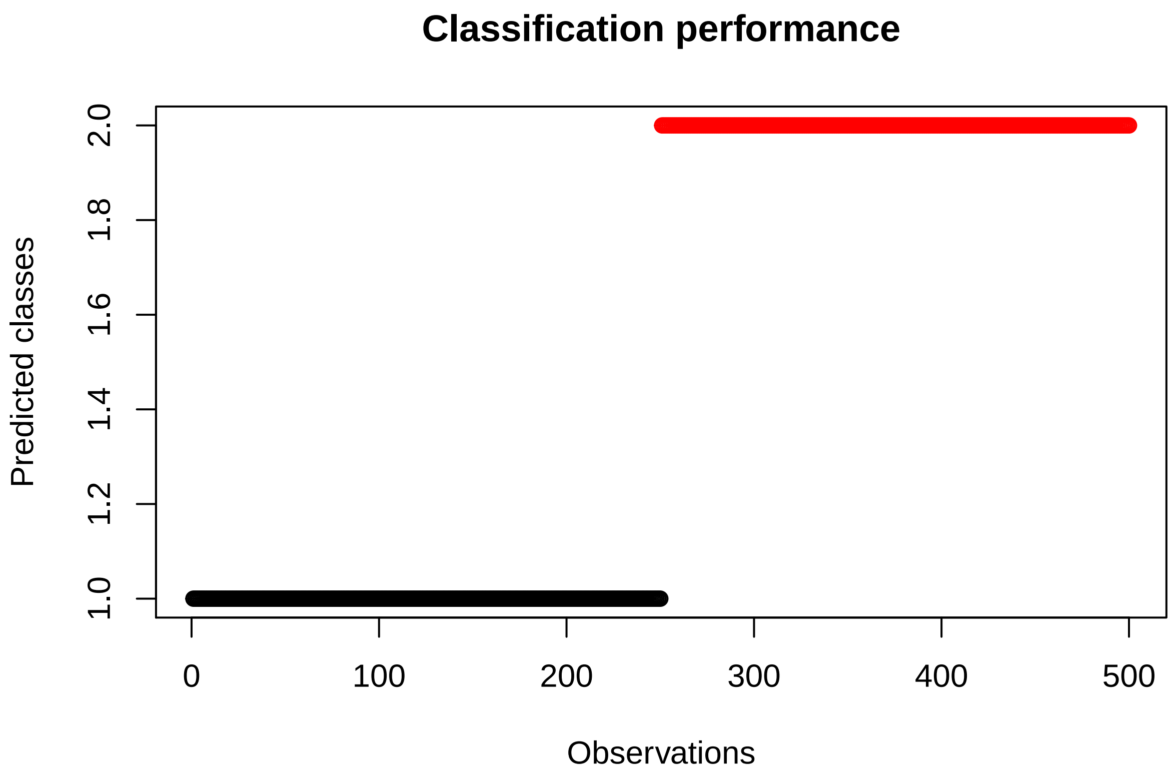
```r
require(PenalizedAD)
data(bcg_data)
```

Given such the data we can use MAES algorithm for model-based clustering to perform classification.

```r
f <- MAES(bcg_data, 2, attr(bcg_data, "label"))
```

and plot the classification results

```r
plot(f[[8]],col=attr(bcg_data, "label"),
     main="Classification performance",ylab="Predicted classes", xlab = "Observations")
```



For the anomaly detection, we make use of the above approach for background density estimation. We can load an example data and fit the anomaly model. The signal is generated by a single Gaussian distribution with mean equal to **0** and diagonal covariance matrix. The experimental data is generated in 90% from the background distribution and in 10% from the signal.

```
data(bcg_data)
data(experiment_data)

#1 fit the background as the potential background model if no signal is to be found
E_bcg=MAES(bcg_data,2,attr(bcg_data, "label"))

#fit the signal

#Iterate between fitting the background on data and fitting the signal on data2
minbic <- Inf
sig <- PAD(E_best=E_bcg, data_exp=experiment_data, gamma_eigen = 1,
            gamma_mean = 4,minbic=minbic,data=bcg_data)

#The component means
sig[[1]]
#>                  [,1]        [,2]        [,3]        [,4]         [,5]
#> background -0.87411826 -0.88431698 -0.87726151 -0.88167680 -0.884283033
#> background  0.87411826  0.88431698  0.87726151  0.88167680  0.884283033
#> signal      0.05656617 -0.04141715 -0.09391226  0.04981984 -0.008932491
#>                  [,6]        [,7]        [,8]      [,9]      [,10] [,11]
#> background -0.89264511 -0.87534006 -0.88073164 0.0000000 0.00000000     0
#> background  0.89264511  0.87534006  0.88073164 0.0000000 0.00000000     0
#> signal     -0.03076101 -0.08770702  0.05119252 0.2014108 0.09837342     0
#>            [,12]      [,13]     [,14] [,15]      [,16]
#> background     0 0.05069119 0.0000000     0 0.00000000
#> background      0 -0.05069119  0.0000000     0  0.00000000
#> signal          0  0.18636343 -0.1352712     0 -0.03733947
```