Package 'rvmbinary'

February 15, 2012

Version 1.0

Date 11-04-2012

Title RVM-Classification and Regression

Author Robert Lowe <ral64@cam.ac.uk>

Maintainer Robert Lowe <ral64@cam.ac.uk>

Depends R (>= 1.8.0), kernlab

Suggests mlbench

Description Performs binary classification and regression of data using a Relevance Vector Machine

License GPL (>= 2)

URL http://www.r-project.org,

Repository CRAN

Date/Publication 2011-10-11 15:24:25

R topics documented:

rvmbinary *Relevance Vector Machine*

Description

The Relevance Vector Machine is a Bayesian model for regression and classification of identical functional form to the support vector machine.

Usage

```
## S3 method for class 'formula'
rvmbinary(formula, data=NULL, ...)

## Default S3 method:
rvmbinary(x,y, kernel="rbfdot", parameters=c(0.1),iterations=100, noisevar=0.1, minmaxdiff = 1e-3, ...

## S3 method for class 'rvmbinary'
print(x, ...)
## S3 method for class 'rvmbinary'
predict(object,newdata, ...)
```

Arguments

formula	Formula interface for rvmbinary
x	The data to be fit by RVM. When not using a formula x can be a matrix or vector containing the training data or a kernel matrix of class rvmkernel of the training data.
data	a data frame containing the variables in the model when using the formula function
y	a response vector with one label for each row/component of x. Can be either a factor (for classification tasks) or a numeric vector (for regression).
kernel	the kernel function used in training and predicting. The AA Kernel is supplied built in which is used by setting the kernel parameter to "aa". The Kernlab kernels are also supplied if Kernlab is installed which provides the most popular kernel functions. These can be used by setting the kernel parameter to the following strings:

- rbfdot Radial Basis kernel "Gaussian"
- polydot Polynomial kernel
- vanilladot Linear kernel
- tanhdot Hyperbolic tangent kernel
- laplacedot Laplacian kernel
- besseldot Bessel kernel
- anovadot ANOVA RBF kernel

- `splinedot` Spline kernel
- `stringdot` String kernel

(default = "rbfdot")

`parameters` a vector of hyper-parameters (kernel parameters). This is a vector which contains the parameters to be used with the kernel function. For valid parameters for existing kernels are :

- `c(sigma)` inverse kernel width for the Radial Basis kernel function "rbfdot" and the Laplacian kernel "laplacedot". And the only parameter for the AA kernel (lambda) which can be set between 0.5-1.0.
- `c(degree, scale, offset)` for the Polynomial kernel "polydot"
- `c(scale, offset)` for the Hyperbolic tangent kernel function "tanhdot"
- `c(sigma, order, degree)` for the Bessel kernel "besseldot".
- `c(sigma, degree)` for the ANOVA kernel "anovadot".
- `c(length, lambda, normalized)` for the "stringdot" kernel where length is the length of the strings considered, lambda the decay factor and normalized a logical parameter determining if the kernel evaluations should be normalized.

(default = c(0.1).

`noisevar` the initial noise variance

`iterations` Number of iterations allowed (default: 100)

`minmaxdiff` termination criteria. Stop when max difference is equal to this parameter (default:1e-3)

`object` The object returned from rvmbinary in order to predict

`newdata` New data for testing

`...` optional parameters to be passed to the low level function `rvmbinary.default`.

Details

The Relevance Vector Machine typically leads to sparser models then the SVM. It is probabilistic by nature and any kernel can be used unlike SVM.

Value

An S4 object of class "rvmbinary" containing the fitted model. Accessor functions can be used to access the slots of the object which include :

`kernel` The kernel used to produce the model

`kernelparameter`
The parameter for the kernel

`nRV` Number of relevance vectors

`type` Classification or Regression

`vectors` The relevance vectors

`values` The weights for the relevance vectors

...

Author(s)

Robert Lowe
<ral64@cam.ac.uk>

References

Tipping, M. E. and A. C. Faul (2003). Fast marginal likelihood maximisation for sparse Bayesian models. In C. M. Bishop and B. J. Frey (Eds.), Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics, Key West, FL, Jan 3-6.

Robert Lowe, Hamse Yussuf Mussa, John Mitchell, and Robert Glen. Classifying Molecules Using a Sparse Probabilistic Kernel Binary Classifier. J. Chem. Inf. Model. DOI: 10.1021/ci200128w

See Also

ksvm

Examples

```
#Classification
data(iris)
#Create training and test splits
datatest=rbind(iris[41:50,],iris[91:100,])
datatrain=rbind(iris[1:40,],iris[51:90,])

#Run model
rvm=rvmbinary(datatrain[,-dim(datatrain)[2]],datatrain[,dim(datatrain)[2]],kernel="rbfdot",parameters=0.1,1000

#Calculate class probability for test set
y=predict(rvm,datatest[,-dim(datatest)[2]])




# create data For REGRESSION
x <- c(seq(-20,-0.1,0.1),seq(0.1,20,0.1))
y <- sin(x)/x + rnorm(400,sd=0.05)

# train relevance vector machine
foo <- rvmbinary(x, y)
foo
# print relevance vectors

# predict and plot
ytest <- predict(foo, x)
plot(x, y, type ="l")
lines(x, ytest, col="red")
```

rvmkernel *Calculate Kernels for RVM Binary*

Description

The Relevance Vector Machine is a Bayesian model for regression and classification of identical functional form to the support vector machine. This method calculates kernels for `rvmbinary`

Usage

```
## S3 method for class 'formula'
rvmkernel(formula, data=NULL, ...)

## Default S3 method:
rvmkernel(x,kernel= "gaus",parameters=c(0.1), ...)
## S3 method for class 'rvmkernel'
print(x, ...)
```

Arguments

`formula`	Formula interface for rvmkernel
`data`	For use with formula interface
`x`	The data to be fit by RVM. When not using a formula x can be a matrix or vector containing the training data
`kernel`	the kernel function used in training and predicting. The AA Kernel is supplied built in which is used by setting the kernel parameter to "aa". The Kernlab kernels are also supplied if Kernlab is installed which provides the most popular kernel functions. These can be used by setting the kernel parameter to the following strings: • `rbfdot` Radial Basis kernel "Gaussian" • `polydot` Polynomial kernel • `vanilladot` Linear kernel • `tanhdot` Hyperbolic tangent kernel • `laplacedot` Laplacian kernel • `besseldot` Bessel kernel • `anovadot` ANOVA RBF kernel • `splinedot` Spline kernel • `stringdot` String kernel (default = "rbfdot")
`parameters`	a vector of hyper-parameters (kernel parameters). This is a vector which contains the parameters to be used with the kernel function. For valid parameters for existing kernels are :

- `c(sigma)` inverse kernel width for the Radial Basis kernel function "rbfdot" and the Laplacian kernel "laplacedot". And the only parameter for the AA kernel (lambda) which can be set between 0.5-1.0.
- `c(degree, scale, offset)` for the Polynomial kernel "polydot"
- `c(scale, offset)` for the Hyperbolic tangent kernel function "tanhdot"
- `c(sigma, order, degree)` for the Bessel kernel "besseldot".
- `c(sigma, degree)` for the ANOVA kernel "anovadot".
- `c(length, lambda, normalized)` for the "stringdot" kernel where length is the length of the strings considered, lambda the decay factor and normalized a logical parameter determining if the kernel evaluations should be normalized.

(default = c(0.1).

`...` optional parameters to be passed to the low level function `rvmkernel.default`.

Details

Produces the kernel necessary for RVM.

Value

An S4 object of class "rvmkernel" containing the calculated kernel.

`BASIS`	The calculated kernel
`type`	The kernel used
`parameter`	The parameter for the kernel
`data`	the training data used

Author(s)

Robert Lowe
<ral64@cam.ac.uk>

References

Tipping, M. E. and A. C. Faul (2003). Fast marginal likelihood maximisation for sparse Bayesian models. In C. M. Bishop and B. J. Frey (Eds.), Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics, Key West, FL, Jan 3-6.

Robert Lowe, Hamse Yussuf Mussa, John Mitchell, and Robert Glen. Classifying Molecules Using a Sparse Probabilistic Kernel Binary Classifier. J. Chem. Inf. Model. DOI: 10.1021/ci200128w

See Also

`ksvm`

Examples

```
# create data
x <- seq(-20,20,0.1)
y <- sin(x)/x + rnorm(401,sd=0.05)

# Calculate the kernel
tmp=rvmkernel(x,kernel="rbfdot")
```

Index