Package 'probsvm'

May 17, 2013

Type Package

Title probsvm: Class probability estimation for Support Vector Machines

Date 2013-5-16

Version 1.00

Author Chong Zhang, Seung Jun Shin, Junhui Wang, Yichao Wu, Hao Helen Zhang, and Yufeng Liu

Maintainer Chong Zhang <chongz@live.unc.edu>

Depends kernlab

Description This package provides multiclass conditional probability estimation for the SVM, which is distributional assumption free.

License GPL-2

LazyLoad yes

R topics documented:

`predict.probsvm` *Prediction function that provides class label prediction and class conditional probability estimation for objects with class "probsvm", returned by the probsvm function.*

Description

Prediction of a test data set on the obtained `probsvm` model. See the description of `probsvm` for details.

Usage

```
## S3 method for class 'probsvm':
predict(object,new.x=NULL,...)
```

Arguments

object	An object returned by a call of probsvm function.
new.x	The new predictor matrix. The number and order of predictors in new.x should be the same as those of x, which is used in the call of probsvm function. If not specified, the program uses the x matrix as the prediction object.
...	Not used.

Value

object	The model from a probsvm function.
new.x	The predictor matrix used for prediction.
pred.prob	The predicted class conditional probability. The class of each column is recorded in the column names.
pred.y	Predicted label for new.x.

See Also

probsvm

---

probsvm *Main function that provides models for multiclass conditional probability estimation and label prediction*

Description

The function uses x and y to build a multiclass prediction model. The multiclass method can be either one-versus-one (ovo), or one-versus-rest (ovr, default). The probability estimation method for a binary classifier is proposed in Wang et al. (2008), and then we use the method proposed in Wu et al. (2004) to combine the results from binary classifiers in the ovo method. For the ovr method, we rescale those binary probabilities for multiclass problems. The function automatically chooses the best penalty parameter via 5-fold (default) Cross Validations (CV). Linear kernel, polynomial kernel and radial(Gaussian) kernel are available. A solution path is provided by Shin et al. (2012), which boosts the computational speed.

Usage

```
probsvm(x, y, fold=5,
kernel=c("linear","polynomial","radial"),
kparam=NULL, Inum=20, type="ovr",
lambdas=2^(-10:10))
```

Arguments

x	The x matrix/data.frame for the training dataset. Columns represent the covariates, and rows represent the instances. There should be no NA/NaN values in x.
y	The labels for the training dataset.
fold	Number of folds in CV. Default 5.

`kernel`	Type of kernel used for learning. `kernel="linear"` for linear kernel (default), `kernel="polynomial"` for polynomial kernel, and `kernel="radial"` for radial(gaussian) kernel.
`kparam`	The parameter for the kernel. For linear kernel, this argument is not needed. In polynomial kernel, it represents the degree of the polynomials, and in radial(gaussian) kernel, it represents the usual sigma value.
`Inum`	Number of knots on [0,1] to estimate the class conditional probability. The larger Inum is, the more accurate the final result is, yet the more time it takes to compute. Default `20`.
`type`	The type of multiclass method. The option `ovo` is for the one-versus-one method, and `ovr` is for the one-versus-rest method (default).
`lambdas`	The user-specified lambda value vector. Each element should be positive. Default `2^(-10:10)`.

Value

`All arguments`	All arguments are returned.
`lambdas`	The lambda values used for selecting the best model. Sorted in an increasing order.
`best.lambda`	The best lambda values selected from CV. Used for model prediction.
`call`	The call of `probsvm`.

Author(s)

Chong Zhang, Seung Jun Shin, Junhui Wang, Yichao Wu, Hao Helen Zhang, and Yufeng Liu

References

Shin, S.J., Y. Wu, and H.H. Zhang (2012). Two-Dimensional Solution Surface for Weighted Support Vector Machines, *Journal of Computational and Graphical Statistics*, in press.

Wang, J., X. Shen, and Y. Liu (2008). Probability estimation for large margin classifiers. *Biometrika* **95(1)**, 149-167.

Wu, T.-F., C.-J. Lin, and R. C. Weng (2004). Probability estimates for multi-class classification by pairwise coupling. *Journal of Machine Learning Research* **5**, 975-1005.

See Also

`predict.probsvm`

Examples

```
# iris data #

data(iris)

iris.x=iris[c(1:20,51:70,101:120),-5]

iris.y=iris[c(1:20,51:70,101:120),5]

iris.test=iris[c(21:50,71:100,121:150),-5]
```

```
a = probsvm(iris.x,iris.y,type="ovo",
Inum=10,fold=2,lambdas=2^seq(-10,10,by=3))
predict(a, iris.test)
```

Index