Package 'PACVB'

February 4, 2016

Type Package

Title Variational Bayes (VB) Approximation of Gibbs Posteriors with Hinge Losses

Version 1.1

Date 2016-01-29

Author James Ridgway

Maintainer James Ridgway <james.ridgway@bristol.ac.uk>

Description Variational Bayesian approximations of Gibbs measures with hinge losses for classification and ranking.

License GPL (>= 2)

Depends Rcpp, MASS

LinkingTo Rcpp, RcppArmadillo, BH

RcppModules GDHingeCxx, GDHingeAUCCxx

NeedsCompilation yes

Repository CRAN

Date/Publication 2016-02-04 15:25:21

R topics documented:

PACVB-package	*Variational Bayes (VB) Approximation of Gibbs Posteriors with Hinge Losses*

Description

The package computes a Gaussian approximation of a Gibbs posterior for convexified risks. We use a hinge loss, and a hinge variation of the AUC loss. The implementation follows the lines of Alquier et al. [2015]. The authors obtain PAC-Bayesian bounds for the variational approximation of an exponential weighted average estimator. The optimal bound is obtain through a gradient descent on a convex objective.

Details

Package:	PACVB
Type:	Package
Version:	1.0
Date:	2015-07-29
License:	GPL (>= 2)

Author(s)

Author: James Ridgway
Maintainer: James Ridgway <james.ridgway@bristol.ac.uk>

References

Alquier, P., Ridgway, J., and Chopin, N. On the properties of variational approximations of Gibbs posteriors. arXiv preprint, 2015

GDHinge	*Computes the VB (variational Bayes) approximation of a Gibbs measure with convexified classification loss using a gradient descent.*

Description

The function computes a VB approximation of a Gibbs measure with a convexified 01-loss using a gradient descent. The user must specify the design matrix, the response vector and the inverse temperature parameter. The number of iteration of the convex solver is fixed a priori (the default value is 10), an informed choice can be made using theorem 6.3 in Alquier et al. [2015].

Usage

```
GDHinge(X,Y,lambda,theta=0,K=100,v=10,ls=FALSE,
                              B=0,family="F1",eps=0.05)
```

Arguments

`X`	Design matrix. The matrix should include a constant column if a bias is to be considered. In addition the Gradient descent has been calibrated considering a centered and scale design matrix.
`Y`	Response vector. The vector should take values in {-1,1}.
`lambda`	Inverse temperature of the Gibbs posterior (See Alquier et al. (2015) for guide lines)
`theta`	Initial value of the gradient descent. In the case of the "F1" family the last value of the vector is the initial log-variance. The vector should be of size p+1, where p is the number of columns of the design matrix. If no initial value are chosen the algorithm is initialized to a Gaussian random vector.
`K`	Number of iteration of the gradient descent. Default value K=10. An informed choice can be made using theorem 6.3 in Alquier et al. [2015].
`v`	Prior variance. The prior is taken to be spherical Gaussian, the variance must therefore be specified in the form of a scalar. For default choices see Alquier et al. [2015]. The default is arbitrarily set to 10.
`ls`	Logical value. Indicates if a linesearch should be used to find an optimal step length. Default value is FALSE. The option is not available for stochastic gradient descent.
`B`	Batch sizes when considering a stochastic gradient descent. B=0 corresponds to standard gradient descent.
`family`	Approximate family to consider when implementing VB. Possible values are: "F0" variance is fixed to 1/(sample size) times identity; "F1" spherical Gaussian. (see Alquier et al. [2015] for details)
`eps`	Probability of the empirical bound of the theoretical risk to be considered. Default is 0.05

Details

The implementation is based on theorem 6.3 of Alquier et al. [2015] using convex solver presented in Nesterov [2004] (section 3.2.3). The calibration depends on an upper bound on the l2 distance between the solution and the initial value. We use an arbitrary value of sqrt (p+1) We also give the possibility to use a linesearch algorithm satisfying the Wolfe conditions.

Value

`m`	Mean of the Gaussian approximation
`s`	Variance of the Gaussian approximation
`bound`	Empirical bound on the aggregated risk. A negative value indicates that temperaure was taken outside of the admissible interval. The bound assumes that each element of the design is bounded by 1. c_x=1 in Alquier et al. [2015]

Warning

The columns of the design matrix should be centered and scale fo proper behaviour of the algorithm.

Author(s)

James Ridgway

References

Alquier, P., Ridgway, J., and Chopin, N. On the properties of variational approximations of Gibbs posteriors. arXiv preprint, 2015.

Nesterov, Y. Introductory lectures on convex optimization, volume 87. Springer Science and Buisness Media, 2004.

Examples

```
data(Pima.tr)
Y<-2*as.matrix(as.numeric(Pima.tr[,8]))-3
X<-data.matrix(Pima.tr[,1:7])
m<-apply(X,2,mean)
v<-apply(X,2,sd)
X<-t(apply(t(apply(X,1,"-",m)),1,"/",v))
X<-cbind(1,X)
l<-45
Sol<-GDHinge(X,Y,l)
```

GDHingeAUC *Compute the VB (variational Bayes) approximation of a Gibbs measure with convexified AUC loss using a gradient descent.*

Description

The function computes a VB approximation of a Gibbs measure with a convexified AUC loss using a gradient descent. The user must specify the design matrix, the response vector and the inverse temperature parameter. The number of iteration of the convex solver is fixed a priori (the default value is 10), the choice can be made using theorem 6.3 in Alquier et al. [2015].

Usage

```
GDHingeAUC(X,Y,lambda,theta=0,K=100,v=10,ls=FALSE,
                                   B=0,family="F1",eps=0.05)
```

Arguments

`X`	Design matrix. The matrix should include a constant column if a bias is to be considered. In addition the Gradient descent has been calibrated considering a centered and scale design matrix.
`Y`	Response vector. The vector should take values in {-1,1}.
`lambda`	Inverse temperature of the Gibbs posterior (See Alquier et al. (2015) for guide lines)
`theta`	Initial value of the gradient descent. In the case of the "F1" family the last value of the vector is the initial log-variance. The vector should be of size p+1, where p is the number of columns of the design matrix. If no initial value are chosen the algorithm is initialized to a Gaussian random vector.
`K`	Number of iteration of the gradient descent. Default value K=10. An informed choice can be made using theorem 6.3 in Alquier et al. [2015].
`v`	Prior variance. The prior is taken to be spherical Gaussian, the variance must therefore be specified in the form of a scalar. For default choices see Alquier et al. [2015]. The default is arbitrarily set to 10.
`ls`	Logical value. Indicates if a line search should be used to find an optimal step length. Default value is FALSE. The option is not available for stochastic gradient descent.
`B`	Batch sizes when considering a stochastic gradient descent. B=0 corresponds to standard gradient descent.
`family`	Approximate family to consider when implementing VB. Possible values are: "F0" variance is fixed to 1/(sample size) times identity; "F1" spherical Gaussian. (see Alquier et al. [2015] for details)
`eps`	Probability of the empirical bound of the theoretical risk to be considered. Default is 0.05

Details

The implementation is based on theorem 6.3 of Alquier et al. [2015] using convex solver presented in Nesterov [2004] (section 3.2.3). The algorithm uses a hinge loss version of AUC and is not explicitly written in Alquier et al. [2015]. The calibration depends on an upper bound on the l2 distance between the solution and the initial value. We use an arbitrary value of sqrt (p+1) We also give the possibility to use a line search algorithm satisfying the Wolfe conditions. We also give the possibility to use a line search algorithm satisfying the Wolfe conditions.

Value

`m`	Mean of the Gaussian approximation
`s`	Variance of the Gaussian approximation
`bound`	Empirical bound on the aggregated risk. A negative value indicates that temperaure was taken outside of the admissible intervale.

Warning

The columns of the design matrix should be centered and scale fo proper behaviour of the algorithm.

Author(s)

James Ridgway

References

Alquier, P., Ridgway, J., and Chopin, N. On the properties of variational approximations of Gibbs posteriors. arXiv preprint, 2015

Nesterov, Y. Introductory lectures on convex optimization, volume 87. Springer Science and Buisness Media, 2004.

Examples

```
data(Pima.tr)
Y<-2*as.matrix(as.numeric(Pima.tr[,8]))-3
X<-data.matrix(Pima.tr[,1:7])
m<-apply(X,2,mean)
v<-apply(X,2,sd)
X<-t(apply(t(apply(X,1,"-",m)),1,"/",v))
l<-45
Sol<-GDHingeAUC(X,Y,l)
```

GDHingeAUCCxx — *C++ internal function to compute the VB approximation with convexified AUC loss (use GDHingeAUC instead).*

Description

The function is the C++ internal function used by GDHingeAUC

Usage

```
GDHingeAUCCxx(...)
```

Arguments

`...`	See arguments of GDHingeAUC

Value

`m`	Mean of the Gaussian approximation
`V`	Variance matrix of the Gaussian approximation

Author(s)

James Ridgway

References

Alquier, P., Ridgway, J., and Chopin, N. On the properties of variational approximations of Gibbs posteriors. arXiv preprint, 2015

`GDHingeCxx` *C++ internal function to compute the VB approximation with convexified classification loss (use GDHinge instead).*

Description

The function is the C++ internal function used by GDHinge

Usage

```
GDHingeCxx(...)
```

Arguments

`...`	See arguments of GDHinge

Value

`m`	Mean of the Gaussian approximation
`V`	Variance matrix of the Gaussian approximation

Author(s)

James Ridgway

References

Alquier, P., Ridgway, J., and Chopin, N. On the properties of variational approximations of Gibbs posteriors. arXiv preprint, 2015

Index