Package ‘MGSDA’

February 19, 2015

Type Package
Title Multi-Group Sparse Discriminant Analysis
Version 1.1
Date 2014-07-07
Author Irina Gaynanova
Maintainer Irina Gaynanova <ig93@cornell.edu>
Imports glmnet, MASS
License GPL (>= 2)
NeedsCompilation yes
Repository CRAN
Date/Publication 2014-08-29 01:35:45

R topics documented:

MGSDA-package ................................. 1
classifyV ................................. 3
cv.dLDA ................................. 5
dLDA ................................. 6

Index

8

Description

Three functions for estimation and prediction: cv.dLDA, dLDA and classifyV.

Author(s)

Irina Gaynanova

Maintainer: Irina Gaynanova <ig93@cornell.edu>

References


Examples

```r
### Example 1
# generate training data
n <- 10
p <- 100
G <- 3
ytrain <- rep(1:G,each=n)
set.seed(1)
xtrain <- matrix(rnorm(p*n*G,n*G,p)
# find matrix of canonical vectors V
V <- dLDA(xtrain,ytrain,lambda=0.1)
sum(rowSums(V)!=0)
# generate test data
m <- 20
set.seed(3)
xtest <- matrix(rnorm(p*m),m,p)
# perform classification
ytest <- classifyV(xtrain,ytrain,xtest,V)
```
classifyV

classifyV(Xtrain, Ytrain, Xtest, V, prior = T, tol1=1e-10)

Arguments

Xtrain
A Nxp data matrix; N observations on the rows and p features on the columns.

Ytrain
A N vector containing the group labels. Should be coded as 1,2,...,G, where G
is the number of groups.

Xtest
A Mxp data matrix; M test observations on the rows and p features on the
columns.

V
A pxr matrix of canonical vectors that is used to classify observations.

prior
A logical indicating whether to put larger weights to the groups of larger size;
the default value is TRUE.

tol1
Tolerance level for the eigenvalues of \(V^tWV\). If some eigenvalues are less than
tol1, the low-rank version of V is used for classification.

Details

For a new observation with the value x, the classification is performed based on the smallest Maha-
lanobis distance in the projected space:

\[
\min_{1 \leq g \leq G} (V^t x - Z_g)(V^t W V)^{-1}(V^t x - Z_g)
\]

where \(Z_g\) are the group-specific means of the training dataset in the projected space and \(W\) is the
sample within-group covariance matrix.

If prior=T, then the above distance is adjusted by \(-2 \log \frac{n_g}{N}\), where \(n_g\) is the size of group g.

Value

Returns a vector of length M with predicted group labels for the test set.

Author(s)

Irina Gaynanova

References

in the p>N setting."

Examples

### Example 1
# generate training data
n=10
p=100
G=3
ytrain=rep(1:G,each=n)
cv.dLDA

```r
set.seed(1)
xtrain=matrix(rnorm(p*n*G),n*G,p)
# find V
V=dLDA(xtrain,ytrain,lambda=0.1)
sum(rowSums(V)!=0)
# generate test data
m=20
set.seed(3)
xtest=matrix(rnorm(p*m),m,p)
# perform classification
ytest=classifyV(xtrain,ytrain,xtest,V)
```

cv.dLDA

Cross-validation for MGSDA

Description

Chooses optimal tuning parameter lambda for function dLDA based on the m-fold cross-validation mean squared error

Usage

```r
cv.dLDA(Xtrain, Ytrain, lambdaval = NULL, nl = 100, msep = 5, eps = 1e-6,
        l_min_ratio = 0.01, myseed=NULL, prior=TRUE)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Xtrain</td>
<td>A Nxp data matrix; N observations on the rows and p features on the columns</td>
</tr>
<tr>
<td>Ytrain</td>
<td>A N vector containing the group labels. Should be coded as 1,2,...,G, where G is the number of groups</td>
</tr>
<tr>
<td>lambdaval</td>
<td>Optional user-supplied sequence of tuning parameters; the default value is NULL and cv.dLDA chooses its own sequence</td>
</tr>
<tr>
<td>nl</td>
<td>Number of lambda values; the default value is 100</td>
</tr>
<tr>
<td>msep</td>
<td>Number of cross-validation folds; the default value is 5</td>
</tr>
<tr>
<td>eps</td>
<td>Tolerance level for the convergence of the optimization algorithm; the default value is 1e-6</td>
</tr>
<tr>
<td>l_min_ratio</td>
<td>Smallest value for lambda, as a fraction of lambda.max, the data-derived value for which all coefficients are zero; the default value is 0.01.</td>
</tr>
<tr>
<td>myseed</td>
<td>Optional specification of random seed for generating the folds; the default value is NULL.</td>
</tr>
<tr>
<td>prior</td>
<td>A logical indicating whether to put larger weights to the groups of larger size; the default value is TRUE.</td>
</tr>
</tbody>
</table>
Value

lambdaval The sequence of tuning parameters used
error The mean cross-validated number of misclassified observations - a vector of length length(lambdaval)
lambda The value of tuning parameter that has the minimal mean cross-validation error
f The mean cross-validated number of non-zero features - a vector of length length(lambdaval)

Author(s)

Irina Gaynanova

References


Examples

```r
### Example 1
n=10
p=100
G=3
ytrain=rep(1:G,each=n)
set.seed(1)
xtrain=matrix(rnorm(p*n*G),n*G,p)
# find optimal tuning parameter
out.cv=cv.dlda(xtrain,ytrain)
# find V
V=dlda(xtrain,ytrain,lambda=out.cv$lambda)
# number of non-zero features
sum(rowSums(V)!=0)
```

---

**dLDA**

_Estimate the matrix of discriminant vectors using L_1 penalty on the rows_

**Description**

Solve Multi-Group Sparse Discriminant Analysis problem for the supplied value of the tuning parameter lambda.

**Usage**

dLDA(xtrain, ytrain, lambda, Vinit = NULL, eps=1e-6)
dLDA

Arguments

- **xtrain**: A Nxp data matrix; N observations on the rows and p features on the columns.
- **ytrain**: A N-vector containing the group labels. Should be coded as 1,2,...,G, where G is the number of groups.
- **lambda**: Tuning parameter.
- **vinit**: A px(G-1) optional initial value for the optimization algorithm; the default value is NULL.
- **eps**: Tolerance level for the convergence of the optimization algorithm; the default value is 1e-6.

Details

Solves the following optimization problem:

\[
\min_{V} \frac{1}{2} \text{Tr}(V^t W V) + \frac{1}{2} \| D^t V - I \|_F^2 + \lambda \sum_{i=1}^{p} \| v_i \|_2
\]

Here W is the within-group sample covariance matrix and D is the matrix of orthogonal contrasts between the group means, both are constructed based on the supplied values of xtrain and ytrain. When \( G = 2 \), the above row penalty reduces to vector L_1 penalty and the function uses glmnet subroutine to solve the optimization problem.

Value

Returns a px(G-1) matrix of canonical vectors V.

Author(s)

Irina Gaynanova

References


Examples

```r
# Example 1
n=10
p=100
G=3
ytrain=rep(1:G,each=n)
set.seed(1)
xtrain=matrix(rnorm(p*n*G),n,G,p)
V=dLDA(xtrain,ytrain,lambda=0.1)
sum(rowSums(V)!=0) # number of non-zero rows
```
Index

classifyV, 3
cv. dLDA, 5
dLDA, 6
MGSDA (MGSDA-package), 1
MGSDA-package, 1