

# Package ‘NetDA’

May 27, 2022

**Type** Package

**Title** Network-Based Discriminant Analysis Subject to Multi-Label Classes

**Version** 0.2.0

**Author** Li-Pang Chen

**Maintainer** Li-Pang Chen <lchen723@nccu.edu.tw>

**Description** Implementation of discriminant analysis with network structures in predictors accommodated to do classification and prediction.

**License** GPL-2

**Encoding** UTF-8

**Imports** glasso

**NeedsCompilation** yes

**LazyData** true

**Suggests** knitr, rmarkdown

**VignetteBuilder** knitr

**Repository** CRAN

**Date/Publication** 2022-05-27 07:10:04 UTC

## R topics documented:

NetDA-package . . . . .	2
Metrics . . . . .	2
NetDA . . . . .	4
WineData . . . . .	5

<b>Index</b>	<b>7</b>
--------------	----------

---

NetDA-package

*Network-Based Discriminant Analysis Subject to Multi-Label Classes*

---

### Description

Implementation of discriminant analysis with network structures in predictors accommodated to do classification and prediction.

### Details

There are two functions in this package: NetDA and Metrics. NetDA aims to construct network-based linear discriminant function and network-based quadratic discriminant function based on the training data, and then do classification for individuals in the testing data. Predicted values can be determined by NetDA. The function Metrics provides a confusion matrix and some commonly used criteria to assess the performance of classification and prediction.

### Author(s)

Chen, L.-P.

Maintainer: Li-Pang Chen <lchen723@nccu.edu.tw>

### References

Chen, L.-P. (2022) Network-Based Discriminant Analysis for Multiclassification. Under revision.

Friedman, J., Hastie, T., and Tibshirani, R. (2008). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9, 432-441.

### See Also

[NetDA](#)

---

Metrics

*Network-Based Discriminant Analysis Subject to Multi-Label Classes*

---

### Description

Implementation of discriminant analysis with network structures in predictors accommodated to do classification and prediction.

### Usage

```
Metrics(yhat, Y_test)
```

**Arguments**

`yhat` an m-dimensional “vector” of the predicted values determined by NetDA.  
`Y_test` an m-dimensional “vector” of the response from the testing data.

**Details**

This function aims to report the performance of classification results. The output includes confusion matrices and some commonly used criteria, such as precision, recall, F-score, and ARI.

**Value**

Confusion matrix  
 A confusion matrix based on predicted values and responses from the testing data  
 (PRE, REC, F-score)  
 Values of precision (PRE), recall (REC), and F-score  
 ARI  
 Values of the adjusted Rand index (ARI)

**Author(s)**

Chen, L.-P.

**References**

Chen, L.-P., Yi, G. Y., Zhang, Q., and He, W. (2019). Multiclass analysis and prediction with network structured covariates. *Journal of Statistical Distributions and Applications*, 6:6.  
 Hubert, L. and Arabie, P. (1985). Comparing partitions. *Journal of Classification*, 2, 193-218.

**See Also**

[NetDA](#)

**Examples**

```
data(WineData)

Y = WineData[,1] # the response
X = WineData[,2:14] # the predictors

D1 = WineData[which(Y==1),]
D2 = WineData[which(Y==2),]
D3 = WineData[which(Y==3),]

Train = rbind(D1[1:45,], D2[1:45,],D3[1:45,]) # user-specific training data
Test = rbind(D1[45:dim(D1)[1],], D2[45:dim(D2)[1],],D3[45:dim(D3)[1],]) # user-specific testing data

X = Train[,2:14]
Y = Train[,1]
X_test = Test[,2:14]
```

```

Y_test = Test[,1]

NetDA(X,Y,method=1,X_test) -> NetLDA
yhat_lda = NetLDA$yhat
Metrics(yhat_lda,Y_test)

#####

NetDA(X,Y,method=2,X_test) -> NetQDA
yhat_qda = NetQDA$yhat
Metrics(yhat_qda,Y_test)

```

---

NetDA

---

*Network-Based Discriminant Analysis Subject to Multi-Label Classes*


---

### Description

Implementation of discriminant analysis with network structures in predictors accommodated to do classification and prediction.

### Usage

```
NetDA(X,Y, method,X_test)
```

### Arguments

X	an (n,p) “matrix” of the predictors from the training data.
Y	an n-dimensional “vector” of the response from the training data.
method	a “scalar” to determine the classification method. “method = 1” represents network-based linear discriminant analysis (NetLDA); “method = 2” represents network-based quadratic discriminant analysis (NetQDA).
X_test	an (m,p) “matrix” of the predictors from the testing data.

### Details

This function is used for the classification using discriminant analysis with network structures in predictors. NetLDA is formulated by linear discriminant function with the corresponding estimated precision matrix obtained by pooling all subjects in the training data; NetLDA is formulated by quadratic discriminant function with the estimated precision matrices determined by stratifying subjects from the associated classes.

### Value

yhat	a vector of predicted responses obtained by NetLDA or NetQDA.
Network	the estimators of confusion matrices.

**Author(s)**

Chen, L.-P.

**References**

Chen, L.-P. (2022) Network-Based Discriminant Analysis for Multiclassification. Under revision.

Friedman, J., Hastie, T., and Tibshirani, R. (2008). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9, 432-441.

**Examples**

```

data(WineData)

Y = WineData[,1]      # the response
X = WineData[,2:14]   # the predictors

D1 = WineData[which(Y==1),]
D2 = WineData[which(Y==2),]
D3 = WineData[which(Y==3),]

Train = rbind(D1[1:45,], D2[1:45,],D3[1:45,]) # user-specific training data
Test = rbind(D1[45:dim(D1)[1],], D2[45:dim(D2)[1],],D3[45:dim(D3)[1],]) # user-specific testing data

X = Train[,2:14]
Y = Train[,1]
X_test = Test[,2:14]
Y_test = Test[,1]

NetDA(X,Y,method=1,X_test) -> NetLDA

yhat_lda = NetLDA$yhat
Net_lda = NetLDA$Network

#####

NetDA(X,Y,method=2,X_test) -> NetQDA

yhat_qda = NetQDA$yhat
Net_qda = NetQDA$Network

```

---

WineData

*Network-Based Discriminant Analysis Subject to Multi-Label Classes*


---

**Description**

The dataset, available at <https://archive.ics.uci.edu/ml/datasets/wine>, illustrates the usage of [NetDA](#) and [Metrics](#) functions. The dataset contains measurements of 178 units where the response is classified as three classes together with 13 relevant predictors.

**Usage**

```
data(WineData)
```

**Format**

The dataset contains one response, 13 predictors, and 178 subjects.

**Examples**

```
Y = WineData[,1]      # the response
X = WineData[,2:14]   # the predictors

D1 = WineData[which(Y==1),]
D2 = WineData[which(Y==2),]
D3 = WineData[which(Y==3),]

Train = rbind(D1[1:45,], D2[1:45,],D3[1:45,]) # user-specific training data
Test = rbind(D1[45:dim(D1)[1],], D2[45:dim(D2)[1],],D3[45:dim(D3)[1],]) # user-specific testing data

X = Train[,2:14]
Y = Train[,1]
X_test = Test[,2:14]
Y_test = Test[,1]
```

# Index

- \* **core**

- NetDA, [4](#)

- \* **datasets**

- WineData, [5](#)

- \* **function**

- Metrics, [2](#)

- NetDA, [4](#)

- \* **package**

- NetDA-package, [2](#)

Metrics, [2](#), [5](#)

NetDA, [2](#), [3](#), [4](#), [5](#)

NetDA-package, [2](#)

WineData, [5](#)