Package ‘esaBcv’

May 29, 2015

Title Estimate Number of Latent Factors and Factor Matrix for Factor Analysis

Version 1.2.1

Description These functions estimate the latent factors of a given matrix, no matter it is high-dimensional or not. It tries to first estimate the number of factors using bi-cross-validation and then estimate the latent factor matrix and the noise variances. For more information about the method, see Art B. Owen and Jingshu Wang 2015 archived article on factor model (http://arxiv.org/abs/1503.03515).

Depends R (>= 3.0.2)

License GPL (>= 2)

LazyData true

Imports corpcor, svd

Suggests MASS

Author Art B. Owen [aut],
Jingshu Wang [aut, cre]

Maintainer Jingshu Wang <wangjingshususan@gmail.com>

NeedsCompilation no

Repository CRAN

Date/Publication 2015-05-29 08:45:53

R topics documented:

ESA 2
EsaBcv 3
esaBcv_package 5
plot.esabcv 6
simdat 7

Index **8**

ESA *Estimate Latent Factor Matrix With Known Number of Factors*

Description

Estimate the latent factor matrix and noise variance using early stopping alternation (ESA) given the number of factors.

Usage

```
ESA(Y, r, X = NULL, center = F, niter = 3, svd.method = "fast")
```

Arguments

Y	observed data matrix. p is the number of variables and n is the sample size. Dimension is `c(n, p)`
r	The number of factors to use
X	the known predictors of size `c(n, k)` if any. Default is NULL (no known predictors). `k` is the number of known covariates.
center	logical, whether to add an intercept term in the model. Default is False.
niter	the number of iterations for ESA. Default is 3.
svd.method	either "fast", "propack" or "standard". "fast" is using the `fast.svd` function in package corpcor to compute SVD, "propack" is using the `propack.svd` to compute SVD and "standard" is using the `svd` function in the base package. Because of PROPACK issues, "propack" fails for some matrices, and when that happens, the function will use "fast" to compute the SVD of that matrix instead. Default method is "fast".

Details

The model used is

$$Y = 1\mu' + X\beta + n^{1/2}UDV' + E\Sigma^{1/2}$$

where D and Σ are diagonal matrices, U and V are orthogonal and μ' and V' mean _mu transposed_ and _V transposed_ respectively. The entries of E are assumed to be i.i.d. standard Gaussian. The model assumes heteroscedastic noises and especially works well for high-dimensional data. The method is based on Owen and Wang (2015). Notice that when nonnull `X` is given or centering the data is required (which is essentially adding a known covariate with all 1), for identifiability, it's required that $< X, U >= 0$ or $< 1, U >= 0$ respectively. Then the method will first make a rotation of the data matrix to remove the known predictors or centers, and then use the latter `n - k` (or `n - k - 1` if centering is required) samples to estimate the latent factors.

Value

The returned value is a list with components

`estSigma`	the diagonal entries of estimated Σ which is a vector of length `p`
`estU`	the estimated U. Dimension `c(n, r)`
`estD`	the estimated diagonal entries of D which is a vector of length `r`
`estV`	the estimated V. Dimension is `c(p, r)`
`beta`	the estimated $beta$ which is a matrix of size `c(k, p)`. Return NULL if the argument `X` is NULL.
`estS`	the estimated signal (factor) matrix S where $$S = 1\mu' + X\beta + n^{1/2}UDV'$$
`mu`	the sample centers of each variable which is a vector of length `p`. It's an estimate of μ. Return NULL if the argument `center` is False.

References

Art B. Owen and Jingshu Wang(2015), Bi-cross-validation for factor analysis, `http://arxiv.org/abs/1503.03515`

Examples

```
Y <- matrix(rnorm(100), nrow = 10) + 3 * rnorm(10) %*% t(rep(1, 10))
ESA(Y, 1)
```

`EsaBcv` *Estimate Latent Factor Matrix*

Description

Find out the best number of factors using Bi-Cross-Validation (BCV) with Early-Stopping-Alternation (ESA) and then estimate the factor matrix.

Usage

```
EsaBcv(Y, X = NULL, r.limit = 20, niter = 3, nRepeat = 12, only.r = F,
  svd.method = "fast", center = F)
```

Arguments

`Y`	observed data matrix. p is the number of variables and n is the sample size. Dimension is `c(n, p)`
`X`	the known predictors of size `c(n, k)` if any. Default is NULL (no known predictors). `k` is the number of known covariates.
`r.limit`	the maximum number of factor to try. Default is 20. Can be set to Inf.

niter	the number of iterations for ESA. Default is 3.
nRepeat	number of repeats of BCV. In other words, the random partition of Y will be repeated for nRepeat times. Default is 12.
only.r	whether only to estimate and return the number of factors.
svd.method	either "fast", "propack" or "standard". "fast" is using the fast.svd function in package corpcor to compute SVD, "propack" is using the propack.svd to compute SVD and "standard" is using the svd function in the base package. Because of PROPACK issues, "propack" fails for some matrices, and when that happens, the function will use "fast" to compute the SVD of that matrix instead. Default method is "fast".
center	logical, whether to add an intercept term in the model. Default is False.

Details

The model is

$$Y = 1\mu' + X\beta + n^{1/2}UDV' + E\Sigma^{1/2}$$

where D and Σ are diagonal matrices, U and V are orthogonal and mu' and V' represent _mu transposed_ and _V transposed_ respectively. The entries of E are assumed to be i.i.d. standard Gaussian. The model assumes heteroscedastic noises and especially works well for high-dimensional data. The method is based on Owen and Wang (2015). Notice that when nonnull X is given or centering the data is required (which is essentially adding a known covariate with all 1), for identifiability, it's required that $< X, U >= 0$ or $< 1, U >= 0$ respectively. Then the method will first make a rotation of the data matrix to remove the known predictors or centers, and then use the latter n - k (or n - k - 1 if centering is required) samples to estimate the latent factors. The rotation idea first appears in Sun et.al. (2012).

Value

EsaBcv returns an obejct of class "esabcv" The function plot plots the cross-validation results and points out the number of factors estimated An object of class "esabcv" is a list containing the following components:

best.r	the best number of factor estimated
estSigma	the diagonal entries of estimated Σ which is a vector of length p
estU	the estimated U. Dimension is c(n, r)
estD	the estimated diagonal entries of D which is a vector of length r
estV	the estimated V. Dimension is c(p, r)
beta	the estimated β which is a matrix of size c(k, p). Return NULL if the argument X is NULL.
estS	the estimated signal(factor) matrix S where $$S = 1\mu' + X\beta + n^{1/2}UDV'$$
mu	the sample centers of each variable which is a vector of length p. It's an estimate of μ. Return NULL if the argument center is False.

`max.r`	the actual maximum number of factors used. For the details of how this is decided, please refer to Owen and Wang (2015)
`result.list`	a matrix with dimension `c(nRepeat, (max.r + 1))` storing the detailed BCV entrywise MSE of each repeat for r from 0 to `max.r`

References

Art B. Owen and Jingshu Wang(2015), Bi-cross-validation for factor analysis, http://arxiv.org/abs/1503.03515

Yunting Sun, Nancy R. Zhang and Art B. Owen, Multiple hypothesis testing adjusted for latent variables, with an application to the AGEMAP gene expression data. The Annuals of Applied Statistics, 6(4): 1664-1688, 2012

See Also

`ESA`, `plot.esabcv`

Examples

```
Y <- matrix(rnorm(100), nrow = 10)
EsaBcv(Y)
```

---

esaBcv_package *esaBcv*

Description

The esaBcv package provides functions to estimate the latent factors of a given matrix, no matter it is high-dimensional or not. It tries to first estimate the number of factors using Bi-cross-validation and then estimate the latent factor matrix and the noise variances using an Early-stopping-alternation method. The method is proposed by Art B. Owen and Jingshu Wang (2015).

Author(s)

Maintainer: Jingshu Wang <wangjingshususan@gmail.com>

See Also

Owen and Wang (2015) Bi-cross-validation for factor analysis, http://arxiv.org/abs/1503.03515

Examples

```
## Not run:
data(simdat)
result <- EsaBcv(simdat$Y)
plot(result)


## End(Not run)
```

---

plot.esabcv *Plot Bi-cross-validation(BCV) Errors*

Description

Plot the average BCV entrywise MSE against the number of factors tried, with error bars and the best number of factors picked.

Usage

```
## S3 method for class 'esabcv'
plot(x, start.r = 0, end.r = NA,
  xlab = "Number of Factors", ylab = "BCV MSE",
  main = "Bi-cross-validation Error", col.line = "BLUE", ...)
```

Arguments

`x`	esabcv object, typically result of `EsaBcv`.
`start.r`	the starting number of factors to display in the plot.
`end.r`	the largest number of factors allowed to display in the plot. Default is NA, which means to make `end.r` as `max.r`.
`xlab`	title for the x axis.
`ylab`	title for the y axis.
`main`	title for the plot.
`col.line`	the line color.
`...`	other parameters to be passed through to plotting functions.

Details

The `esabcv` object contains the raw BCV result `result.list`, which is a matrix with dimension `c(nRepeat, (max.r + 1))` where `nRepeat` is the number of BCV repeats and `max.r` is the maximum number of factors tried. If either tail of the error curve dominates, then the user has the option to change the start and end rank for plotting.

Value

A plot ploting the average BCV entrywise MSE against the number of factors tried (start.r to `max.r + 1`), with error bars (one standard deviation) in grey and selected number of factors marked by a red crossing.

Examples

```
## Not run:
data(simdat)
result <- EsaBcv(simdat$Y)
plot(result)
plot(result, start.r = 1)


## End(Not run)
```

`simdat` *Example Dataset*

Description

The data is a simulated data set where the data matrix is generated from the latent factor model

$$Y = n^{1/2}UDV' + E\Sigma^{1/2}$$

where D and Σ are diagonal matrices, and U and V are orthogonal. V' means _V transposed_. For the factors, we include one giant factor, five useful factors, one harmful factor and one undetectable factor. For more details of the simulation method used, please refer to Appendix A.1 of Owen and Wang (2015) Bi-cross-validation for factor analysis, http://arxiv.org/abs/1503.03515.

Details

The dataset is a list of components:

- `Y` a data matrix of 200 by 1000, where each row is a sample and each column is a variable
- `U` the orthogonal factor matrix U of size 200 by 8.
- `V` the orthogonal factor matrix V of size 1000 by 8.
- `D` the vector of diagonal entries of D.
- `Sigma` the vector of diagonal entries of Σ.
- `oracle.r` the oracle rank (the optimal number of factors that should be kept) of the factor matrix.

Index

class, *4*

ESA, 2, *5*
EsaBcv, 3, *6*
esaBcv_package, 5
esaBcv_package-package
    (esaBcv_package), 5

fast.svd, *2*, *4*

plot.esabcv, *5*, 6
propack.svd, *2*, *4*

simdat, 7
svd, *2*, *4*