Package 'nsprcomp'

October 13, 2022

Version 0.5.1-2

Title Non-Negative and Sparse PCA

Description Two methods for performing a constrained principal component analysis (PCA), where non-negativity and/or sparsity constraints are enforced on the principal axes (PAs). The function 'nsprcomp' computes one principal component (PC) after the other. Each PA is optimized such that the corresponding PC has maximum additional variance not explained by the previous components. In contrast, the function 'nscumcomp' jointly computes all PCs such that the cumulative variance is maximal. Both functions have the same interface as the 'prcomp' function from the 'stats' package (plus some extra parameters), and both return the result of the analysis as an object of class 'nsprcomp', which inherits from 'prcomp'. See <https://sigg-iten.ch/learningbits/2013/05/27/nsprcomp-is-on-cran/> and Sigg et al. (2008) <doi:10.1145/1390156.1390277> for more details.

URL https://sigg-iten.ch/research/

BugReports https://github.com/chrsigg/nsprcomp/issues

License GPL (>= 2)

Depends R (>= 3.4.0)

Imports stats

Suggests MASS, testthat (>= 0.8), roxygen2

RoxygenNote 6.0.1

NeedsCompilation no

Author Christian Sigg [aut, cre] (<https://orcid.org/0000-0003-1067-9224>), R Core team [ctb] (prcomp interface, formula implementation and documentation)

Maintainer Christian Sigg <christian@sigg-iten.ch>

Repository CRAN

Date/Publication 2018-06-05 11:48:17 UTC

R topics documented:

`asdev` *Additional Explained Standard Deviation*

Description

`asdev` computes the *additional* standard deviation explained by each principal component, taking into account the possible non-orthogonality of the pseudo-rotation matrix $\mathbf{W}$.

Usage

```
asdev(x, w, center = TRUE, scale. = FALSE)
```

Arguments

`x`	a numeric data matrix with the observations as rows
`w`	a numeric data matrix with the principal axes as columns
`center`	a logical value indicating whether the empirical mean of `x` should be subtracted. Alternatively, a vector of length equal to the number of columns of `x` can be supplied. The value is passed to `scale`.
`scale.`	a logical value indicating whether the columns of `x` should be scaled to have unit variance before the analysis takes place. The default is `FALSE` for consistency with `prcomp`. Alternatively, a vector of length equal to the number of columns of `x` can be supplied. The value is passed to `scale`.

Details

The additional standard deviation of a component is measured after projecting the corresponding principal axis to the ortho-complement space spanned by the previous principal axes. This procedure ensures that the variance explained by non-orthogonal principal axes is not counted multiple times. If the principal axes are pairwise orthogonal (e.g. computed using standard PCA), the additional standard deviations are identical to the standard deviations of the columns of the scores matrix $\mathbf{XW}$.

`asdev` is also useful to build a partial PCA model from $\mathbf{W}$, to be completed with additional components computed using `nsprcomp`.

Value

`asdev` returns a list with class `(nsprcomp, prcomp)` containing the following elements:

`sdev`	the additional standard deviation explained by each component
`rotation`	copied from the input argument `w`
`x`	the scores matrix $\mathbf{XW}$, containing the principal components as columns (after centering and scaling if requested)
`center, scale.`	the centering and scaling used
`xp`	the deflated data matrix corresponding to `x`
`q`	an orthonormal basis for the principal subspace

Note

The PCA terminology is not consistent across the literature. Given a zero mean data matrix $\mathbf{X}$ (with observations as rows) and a basis $\mathbf{W}$ of the principal subspace, we define the scores matrix as $\mathbf{Z} = \mathbf{XW}$ which contains the principal components as its columns. The columns of the pseudo-rotation matrix $\mathbf{W}$ are called the principal axes, and the elements of $\mathbf{W}$ are called the loadings.

References

Mackey, L. (2009) Deflation Methods for Sparse PCA. In *Advances in Neural Information Processing Systems* (pp. 1017–1024).

`cardinality` *Cardinality of Column Vectors*

Description

Computes the cardinality (the sum of non-zero elements) of each column of the matrix $\mathbf{W}$.

Usage

```
cardinality(w)
```

Arguments

`w`	a numeric matrix, e.g. the rotation matrix of a sparse PCA analysis

nscumcomp *Non-Negative and Sparse Cumulative PCA*

Description

Performs a PCA-like analysis on the given data matrix, where non-negativity and/or sparsity constraints are enforced on the principal axes (PAs). In contrast to regular PCA, which greedily maximises the variance of each principal component (PC), nscumcomp *jointly* optimizes the components such that the cumulative variance of all PCs is maximal.

Usage

```
nscumcomp(x, ...)

## Default S3 method:
nscumcomp(x, ncomp = min(dim(x)), omega = rep(1, nrow(x)),
  k = d * ncomp, nneg = FALSE, gamma = 0, center = TRUE,
  scale. = FALSE, nrestart = 5, em_tol = 0.001, em_maxiter = 20,
  verbosity = 0, ...)

## S3 method for class 'formula'
nscumcomp(formula, data = NULL, subset, na.action, ...)
```

Arguments

x	a numeric matrix or data frame which provides the data for the analysis.
...	arguments passed to or from other methods.
ncomp	the number of principal components (PCs) to be computed. The default is to compute a full basis for x.
omega	a vector with as many entries as there are data samples, to perform weighted PCA (analogous to weighted least-squares regression). The default is an equal weighting of all samples.
k	an upper bound on the total number of non-zero loadings of the pseudo-rotation matrix $\mathbf{W}$. k is increased if necessary to ensure at least one non-zero coefficient per principal axis.
nneg	a logical value indicating whether the loadings should be non-negative, i.e. the PAs should be constrained to the non-negative orthant.
gamma	a non-negative penalty on the divergence from orthonormality of the pseudo-rotation matrix. The default is not to penalize, but a positive value is sometimes necessary to avoid PAs collapsing onto each other.
center	a logical value indicating whether the empirical mean of (the columns of) x should be subtracted. Alternatively, a vector of length equal to the number of columns of x can be supplied. The value is passed to scale.

scale.	a logical value indicating whether the columns of x should be scaled to have unit variance before the analysis takes place. The default is FALSE for consistency with prcomp. Alternatively, a vector of length equal to the number of columns of x can be supplied. The value is passed to scale.
nrestart	the number of random restarts for computing the pseudo-rotation matrix via expectation-maximization (EM) iterations. The solution achieving the minimum of the objective function over all random restarts is kept. A value greater than one can help to avoid poor local minima.
em_tol	If the relative change of the objective is less than em_tol between iterations, the EM procedure is asssumed to have converged to a local optimum.
em_maxiter	the maximum number of EM iterations to be performed. The EM procedure is terminated if either the em_tol or the em_maxiter criterion is satisfied.
verbosity	an integer specifying the verbosity level. Greater values result in more output, the default is to be quiet.
formula	a formula with no response variable, referring only to numeric variables.
data	an optional data frame (or similar: see model.frame) containing the variables in the formula formula. By default the variables are taken from environment(formula).
subset	an optional vector used to select rows (observations) of the data matrix x.
na.action	a function which indicates what should happen when the data contain NAs. The default is set by the na.action setting of options, and is na.fail if that is unset.

Details

nscumcomp computes all PCs jointly using expectation-maximization (EM) iterations. The M-step is equivalent to minimizing the objective function

$$\left\|\mathbf{X} - \mathbf{Z}\mathbf{W}^\top\right\|_F^2 + \gamma \left\|\mathbf{W}^\top\mathbf{W} - \mathbf{I}\right\|_F^2$$

w.r.t. the pseudo-rotation matrix $\mathbf{W}$, where $\mathbf{Z} = \mathbf{X}\mathbf{W}\left(\mathbf{W}^\top\mathbf{W}\right)^{-1}$ is the scores matrix modified to account for the non-orthogonality of $\mathbf{W}$, $\mathbf{I}$ is the identity matrix and gamma is the Lagrange parameter associated with the ortho-normality penalty on $\mathbf{W}$. Non-negativity of the loadings is achieved by enforcing a zero lower bound in the L-BFGS-B algorithm used for the minimization of the objective, and sparsity is achieved by a subsequent soft thresholding of $\mathbf{W}$.

Value

nscumcomp returns a list with class (nsprcomp, prcomp) containing the following elements:

sdev	the additional standard deviation explained by each component, see asdev.
rotation	the matrix of non-negative and/or sparse loadings, containing the principal axes as columns.
x	the scores matrix $\mathbf{X}\mathbf{W}$ containing the principal components as columns (after centering and scaling if requested)
center, scale	the centering and scaling used, or FALSE

xp	the deflated data matrix corresponding to x
q	an orthonormal basis for the principal subspace

The components are returned in order of decreasing variance for convenience.

Note

The PCA terminology is not consistent across the literature. Given a zero mean data matrix $\mathbf{X}$ (with observations as rows) and a basis $\mathbf{W}$ of the principal subspace, we define the scores matrix as $\mathbf{Z} = \mathbf{XW}$ which contains the principal components as its columns. The columns of the pseudo-rotation matrix $\mathbf{W}$ are called the principal axes, and the elements of $\mathbf{W}$ are called the loadings.

See Also

asdev, peav, nsprcomp, scale

Examples

```
if (requireNamespace("MASS", quietly = TRUE)) withAutoprint({

  set.seed(1)

  # Regular PCA, with tolerance set to return five PCs
  pca <- prcomp(MASS::Boston, tol = 0.35, scale. = TRUE)
  cumsum(pca$sdev[1:5])

  # Sparse cumulative PCA with five components and a total of 20 non-zero loadings.
  # The orthonormality penalty is set to a value which avoids co-linear principal
  # axes. Note that the non-zero loadings are not distributed uniformly over
  # the components.
  scc <- nscumcomp(MASS::Boston, ncomp = 5, k = 20, gamma = 1e4, scale. = TRUE)
  cumsum(scc$sdev)
  cardinality(scc$rotation)

  # Non-negative sparse cumulative PCA
  nscumcomp(MASS::Boston, ncomp = 5, nneg = TRUE, k = 20, gamma = 1e4, scale. = TRUE)
})
```

---

nsprcomp *Non-Negative and Sparse PCA*

Description

Performs a constrained principal component analysis, where non-negativity and/or sparsity constraints are enforced on the principal axes. The result is returned as an object of class nsprcomp, which inherits from prcomp.

Usage

```
nsprcomp(x, ...)

## Default S3 method:
nsprcomp(x, retx = TRUE, ncomp = min(dim(x)),
  omega = rep(1, nrow(x)), k = ncol(x), nneg = FALSE, center = TRUE,
  scale. = FALSE, tol = NULL, nrestart = 5, em_tol = 0.001,
  em_maxiter = 100, partial_model = NULL, verbosity = 0, ...)

## S3 method for class 'formula'
nsprcomp(formula, data = NULL, subset, na.action, ...)
```

Arguments

x	a numeric matrix or data frame which provides the data for the principal component analysis.
...	arguments passed to or from other methods.
retx	a logical value indicating whether the principal components, i.e. x projected into the principal subspace, should be returned.
ncomp	the number of principal components (PCs) to be computed. With the default setting, PCs are computed until x is fully deflated. ncomp can be specified implicitly if k is given as a vector.
omega	a vector with as many entries as there are data samples, to perform weighted PCA (analogous to weighted least-squares regression). The default is an equal weighting of all samples.
k	either a scalar or a vector of length ncomp, specifying the upper bounds on the cardinalities of the principal axes (PAs).
nneg	a logical value indicating whether the loadings should be non-negative, i.e. the PAs should be constrained to the non-negative orthant.
center	a logical value indicating whether the empirical mean of (the columns) of x should be subtracted. Alternatively, a vector of length equal to the number of columns of x can be supplied. The value is passed to scale.
scale.	a logical value indicating whether the columns of x should be scaled to have unit variance before the analysis takes place. The default is FALSE for consistency with prcomp. Alternatively, a vector of length equal to the number of columns of x can be supplied. The value is passed to scale.
tol	a threshold indicating the magnitude below which components should be omitted. Components are omitted if their standard deviations are less than or equal to tol times the standard deviation of the first component. With the default NULL setting, no components are omitted. With tol = 0 or tol = sqrt(.Machine$double.eps), essentially constant components are omitted.
nrestart	the number of random restarts for computing the principal component via expectation-maximization (EM) iterations. The solution achieving maximum standard deviation over all random restarts is kept. A value greater than one can help to avoid poor local maxima.

em_tol	If the relative change of the objective is less than em_tol between iterations, the EM procedure is asssumed to have converged to a local optimum.
em_maxiter	the maximum number of EM iterations to be performed. The EM procedure is terminated if either the em_tol or the em_maxiter criterion is satisfied.
partial_model	NULL or an object of class nsprcomp. The computation can be continued from a partial model by providing a nsprcomp object (either from a previous run of this function or from asdev) and setting ncomp to a value greater than the number of components contained in the partial model. See the examples for an illustration.
verbosity	an integer specifying the verbosity level. Greater values result in more output, the default is to be quiet.
formula	a formula with no response variable, referring only to numeric variables.
data	an optional data frame (or similar: see model.frame) containing the variables in the formula. By default the variables are taken from environment(formula).
subset	an optional vector used to select rows (observations) of the data matrix x.
na.action	a function which indicates what should happen when the data contain NAs. The default is set by the na.action setting of options, and is na.fail if that is unset. The 'factory-fresh' default is na.omit.

Details

nsprcomp computes a principal component (PC) using expectation-maximization iterations, where non-negativity of the loadings is achieved by projecting the principal axis (PA) into the non-negative orthant, and sparsity of the loadings is achieved by soft thresholding (Sigg and Buhmann, 2008).

Because constrained principal axes no longer correspond to true eigenvectors of the covariance matrix and are usually not pairwise orthogonal, special attention needs to be paid when computing more than a single PC. The algorithm implements the generalized deflation method proposed by Mackey (2009) to maximize the additional variance of each component. Given a basis of the space spanned by the previous PAs, the variance of the PC is maximized after projecting the current PA to the ortho-complement space of the basis. This procedure maximizes the additional variance not explained by previous components, and is identical to standard PCA if no sparsity or non-negativity constraints are enforced on the PAs.

See the references for further details.

Value

nsprcomp returns a list with class (nsprcomp, prcomp) containing the following elements:

sdev	the additional standard deviation explained by each component, see asdev.
rotation	the matrix of non-negative and/or sparse loadings, containing the principal axes as columns.
x	the scores matrix $\mathbf{XW}$ containing the principal components as columns (after centering and scaling if requested). For the formula method, napredict is applied to handle the treatment of values omitted by the na.action.
center, scale	the centering and scaling used, or FALSE
xp	the deflated data matrix corresponding to x
q	an orthonormal basis for the principal subspace

Note

The PCA terminology is not consistent across the literature. Given a zero mean data matrix $\mathbf{X}$ (with observations as rows) and a basis $\mathbf{W}$ of the principal subspace, we define the scores matrix as $\mathbf{Z} = \mathbf{XW}$ which contains the principal components as its columns. The columns of the pseudo-rotation matrix $\mathbf{W}$ are called the principal axes, and the elements of $\mathbf{W}$ are called the loadings.

Deflating the data matrix accumulates numerical errors over successive PCs.

References

Sigg, C. D. and Buhmann, J. M. (2008) Expectation-Maximization for Sparse and Non-Negative PCA. In *Proceedings of the 25th International Conference on Machine Learning* (pp. 960–967).

Mackey, L. (2009) Deflation Methods for Sparse PCA. In *Advances in Neural Information Processing Systems* (pp. 1017–1024).

See Also

asdev, peav, prcomp, scale

Examples

```
if (requireNamespace("MASS", quietly = TRUE)) withAutoprint({
  set.seed(1)

  # Regular PCA, with the tolerance set to return five PCs
  prcomp(MASS::Boston, tol = 0.36, scale. = TRUE)

  # Sparse PCA with different cardinalities per component. The number of components
  # is derived from the length of vector k.
  nsprcomp(MASS::Boston, k = c(13, 7, 5, 5, 5), scale. = TRUE)

  # Non-negative sparse PCA with four components. Note that the principal axes
  # naturally have a high degree of orthogonality, because each component
  # maximizes the additional variance not already explained.
  set.seed(1)
  nsprcomp(MASS::Boston, k = c(7, 5, 2, 2), nneg = TRUE, scale. = TRUE)

  # The optimization can get stuck in local optima. Increase the number of
  # random restarts or the number of power iterations to likely obtain decreasing
  # standard deviations.
  set.seed(1)
  (nspc <- nsprcomp(MASS::Boston, k = c(7, 5, 2, 2), nneg = TRUE, scale. = TRUE,
                    nrestart = 10, em_tol = 1e-4, verbosity = 1))

  # continue the computation of components from a partial model
 nsprcomp(MASS::Boston, k = 3, ncomp = 5, nneg = TRUE, scale. = TRUE, partial_model = nspc)

  # The reconstruction error for each sample can be influenced using the
  # weighting vector omega. To reconstruct the data, the generalized
  # inverse of the pseudo-rotation matrix has to be used, because the constrained
  # principal axes are in general not pairwise orthogonal.
```

```
  set.seed(1)
  X <- matrix(runif(5*10), 5)
  nspc <- nsprcomp(X, omega = c(5, 1, 1, 1, 5), ncomp = 2, nneg = TRUE)
  X_hat <- predict(nspc)%*%MASS::ginv(nspc$rotation) + matrix(1,5,1)%*%nspc$center
  rowSums((X - X_hat)^2)
})
```

---

peav *Percentage Explained Additional Variance*

Description

peav computes the percentage of the explained _additional_ variance of each principal component, taking into account the possible non-orthogonality of the pseudo-rotation matrix $\mathbf{W}$.

Usage

```
peav(x, w, center = TRUE, scale. = FALSE)
```

Arguments

x	a numeric data matrix with the observations as rows
w	a numeric data matrix with the principal axes as columns
center	a logical value indicating whether the empirical mean of x should be subtracted. Alternatively, a vector of length equal to the number of columns of x can be supplied. The value is passed to scale.
scale.	a logical value indicating whether the columns of x should be scaled to have unit variance before the analysis takes place. The default is FALSE for consistency with prcomp. Alternatively, a vector of length equal to the number of columns of x can be supplied. The value is passed to scale.

Details

The explained additional variance is computed using asdev and divided by the total variance of the data to obtain percentages. sum(peav(x, w)) is equal to one if $\mathbf{W}$ is an orthonormal basis, e.g. the rotation matrix of a standard PCA.

peav is useful to compare the solutions of various constrained PCA methods w.r.t. standard PCA.

Note

The method produces different results than the "percentage explained variance" (pev) computed by the spca function from the elasticnet package.

See Also

asdev, scale

Index