Package ‘svd’

February 20, 2015

Type Package
Title Interfaces to various state-of-art SVD and eigensolvers
Version 0.3.3-2
Date 2013-04-14
Author Anton Korobeynikov
Maintainer Anton Korobeynikov <anton@korobeynikov.info>
Description This package provides various R bindings to various SVD and eigensolvers (PROPACK, nuTRLan)
License BSD_3_clause + file LICENSE
URL http://github.com/asl/svd
NeedsCompilation yes
Repository CRAN
Date/Publication 2014-03-07 11:58:49

R topics documented:

extmat ................................................................. 1
svd ................................................................. 3

Index

extmat External matrices operations.

Description

A set of routines to operate on "external" matrices.
Usage

is.extmat(X)
extmat.ncol(X)
extmat.nrow(X)
extmat(mul, tmul, nrow, ncol, env = .GlobalEnv)
ematmul(emat, v, transposed = FALSE)

Arguments

X, emat matrix to operate on
mul function performing the multiplication of matrix to vector
tmul function performing the multiplication of transposed matrix to vector
nrow number of rows of the matrix
ncol number of columns of the matrix
env environment, where matrix-vector multiplication function call is evaluated in
transposed logical, if 'TRUE' the multiplication is performed with the transposed matrix.
v vector to multiply with.

Details

These routines checks whether the given external pointer actually points to "external matrix" structure and allow to extract the number of columns and rows respectively.

'extmat' is a convenient wrapper which allows one provide just the routines which will multiply with matrix and the transposed one (e.g. if the matrix is sparse or structured) and allow to use the SVD routines of the package

Examples

## Not run:
library(Matrix)
f <- function(v) as.numeric(A %*% v) # Convert Matrix object back to vector
tf <- function(v) as.numeric(t(A) %*% v) # Convert Matrix object back to vector

e <- new.env()
assign("A", USCounties, e)
assign("tA", t(USCounties), e)
environment(f) <- e
environment(tf) <- e

m <- extmat(f, tf, nrow(USCounties), ncol(USCounties))
system.time(v1 <- propack.svd(m, neig = 10))
# user system elapsed
# 0.252 0.007 0.259
system.time(v2 <- propack.svd(as.matrix(USCounties), neig = 10))
# user system elapsed
# 8.563 0.027 8.590

## End(Not run)
**svd**

*Generic Singular Value Decomposition of a Matrix*

**Description**

Compute the singular-value decomposition of a real rectangular matrix.

**Usage**

```r
propack.svd(x, neig = min(m, n), opts = list())
trlan.svd(x, neig = min(m, n), opts = list(), lambda = NULL, U = NULL)
```

**Arguments**

- `x`: the matrix to be decomposed. This can be either normal matrix or 'external matrix' object (e.g. one, created via 'new.hmat' function).
- `neig`: number of desired eigentriples
- `opts`: different options for eigensolver. See 'Details' section for more information
- `lambda`: set of already computed singular values (used for continuation of the decomposition).
- `U`: matrix of already computed eigenvectors (used for continuation of the decomposition).

**Details**

These routines provide an interface to two state-of-art implementations of truncated SVD.

PROPACK does this via the implicitly restarted Lanczos bidiagonalization with partial reorthogonalization. nu-TRLAN does the thick-restart Lanczos eigendecomposition of cross-product matrix.

'opts' is a list of different options which can be passed to the routines. Note that by default more or less suitable values for these options are set by the routines automatically.

The options for PROPACK are:

- `kmax`: integer, maximum number of iterations.
- `dim`: integer, dimension of Krylov subspace.
- `p`: integer, number of shifts per restart.
- `maxiter`: integer, maximum number of restarts.
- `tol`: numeric, tolerance level.
- `verbose`: logical, if 'TRUE', provide verbose output.

The options for nu-TRLAN are:

- `kmax`: integer, maximum number of iterations.
- `maxiter`: integer, maximum number of matrix-vector products.
- `tol`: numeric, tolerance level.
- `verbose`: integer, verboxeness level.
Value

The returned value is a list with components

- **d**: a vector containing the singular values of 'x'
- **u**: a matrix whose columns contain the left singular vectors of 'X'
- **v**: a matrix whose columns contain the right singular vectors of 'X' (only for 'propack.svd')

References


Index

*Topic algebra
  svd, 3
*Topic array
  svd, 3

ematmul (extmat), 1
extmat, 1
is.extmat (extmat), 1
propack.svd (svd), 3
svd, 3
trlan.eigen (svd), 3
trlan.svd (svd), 3