

# Package ‘ROptSpace’

February 21, 2020

**Type** Package

**Title** Matrix Reconstruction from a Few Entries

**Version** 0.2.2

**Description** Matrix reconstruction, also known as matrix completion, is the task of inferring missing entries of a partially observed matrix. This package provides a method called OptSpace, which was proposed by Keshavan, R.H., Oh, S., and Montanari, A. (2009) <doi:10.1109/ISIT.2009.5205567> for a case under low-rank assumption.

**License** GPL (>= 3)

**Encoding** UTF-8

**LazyData** true

**Imports** Rcpp, Rdpack, stats, utils

**LinkingTo** Rcpp, RcppArmadillo

**RdMacros** Rdpack

**RoxygenNote** 7.0.2

**NeedsCompilation** yes

**Author** Kisung You [aut, cre] (<<https://orcid.org/0000-0002-8584-459X>>)

**Maintainer** Kisung You <[kyoustat@gmail.com](mailto:kyoustat@gmail.com)>

**Repository** CRAN

**Date/Publication** 2020-02-21 18:10:05 UTC

## R topics documented:

OptSpace . . . . .	2
ROptSpace . . . . .	3

<b>Index</b>	<b>4</b>
--------------	----------

---

OptSpace	<i>OptSpace : an algorithm for matrix reconstruction from a partially revealed set</i>
----------	--

---

### Description

Let's assume an ideal matrix  $M$  with  $(m \times n)$  entries with rank  $r$  and we are given a partially observed matrix  $M_E$  which contains many missing entries. Matrix reconstruction - or completion - is the task of filling in such entries. OptSpace is an efficient algorithm that reconstructs  $M$  from  $|E| = O(rn)$  observed elements with relative root mean square error (RMSE)

$$RMSE \leq C(\alpha)\sqrt{nr/|E|}$$

### Usage

```
OptSpace(A, ropt = NA, niter = 50, tol = 1e-06, showprogress = TRUE)
```

### Arguments

**A** an  $(n \times m)$  matrix whose missing entries should be flagged as NA.  
**ropt** NA to guess the rank, or a positive integer as a pre-defined rank.  
**niter** maximum number of iterations allowed.  
**tol** stopping criterion for reconstruction in Frobenius norm.  
**showprogress** a logical value; TRUE to show progress, FALSE otherwise.

### Value

a named list containing

- X** an  $(n \times r)$  matrix as left singular vectors.
- S** an  $(r \times r)$  matrix as singular values.
- Y** an  $(m \times r)$  matrix as right singular vectors.
- dist** a vector containing reconstruction errors at each successive iteration.

### References

Keshavan RH, Montanari A, Oh S (2010). "Matrix Completion From a Few Entries." *IEEE Transactions on Information Theory*, **56**(6), 2980–2998. ISSN 0018-9448, doi: [10.1109/TIT.2010.2046205](https://doi.org/10.1109/TIT.2010.2046205), <http://ieeexplore.ieee.org/document/5466511/>.

**Examples**

```

## Parameter Settings
n = 1000;
m = 100;
r = 3;
tolerance = 1e-7
eps = 10*r*log10(n)

## Generate a matrix with given data
U = matrix(rnorm(n*r),nrow=n)
V = matrix(rnorm(m*r),nrow=m)
Sig = diag(r)
M0 = U%%Sig%%t(V)

## Set some entries to be NA with probability eps/sqrt(m*n)
E = 1 - ceiling(matrix(rnorm(n*m),nrow=n) - eps/sqrt(m*n))
M_E = M0
M_E[(E==0)] = NA

## Create a noisy version
noiselevel = 0.1
M_E_noise = M_E + matrix(rnorm(n*m),nrow=n)*noiselevel

## Use OptSpace for reconstruction
res1 = OptSpace(M_E,tol=tolerance)
res2 = OptSpace(M_E_noise,tol=tolerance)

## Compute errors for both cases using Frobenius norm
err_clean = norm(res1$X%%res1$S%%t(res1$Y)-M0, 'f')/sqrt(m*n)
err_noise = norm(res2$X%%res2$S%%t(res2$Y)-M0, 'f')/sqrt(m*n)

## print out the results
m1 = sprintf('RMSE without noise      : %e',err_clean)
m2 = sprintf('RMSE with noise of %.2f  : %e',noiselevel,err_noise)
print(m1)
print(m2)

```

**Description**

Matrix reconstruction, also known as matrix completion, is the task of inferring missing entries of a partially observed matrix. This package provides a method called OptSpace, which was proposed by Keshavan, R.H., Oh, S., and Montanari, A. (2009) <doi:10.1109/ISIT.2009.5205567> for a case under low-rank assumption.

# Index

OptSpace, [2](#)

package-ROptSpace (ROptSpace), [3](#)

ROptSpace, [3](#)