Package ‘R1magic’
August 29, 2016

Type    Package
Title   Compressive Sampling: Sparse Signal Recovery Utilities
Version 0.3.2
Date    2015-04-19
Maintainer Mehmet Suzen <mehmet.suzen@physics.org>
Depends stats, utils
Description Utilities for sparse signal recovery suitable for compressed sensing. L1, L2 and TV penalties, DFT basis matrix, simple sparse signal generator, mutual cumulative coherence between two matrices and examples, Lp complex norm, scaling back regression coefficients.
License GPL (>= 3)
LazyLoad yes
NeedsCompilation no
Author Mehmet Suzen [aut, cre]
Repository CRAN
Date/Publication 2015-04-19 22:18:58

**R topics documented:**

- R1magic-package
- CompareL1_L2_TV1
- DFTMatrix0
- DFTMatrixPlain
- GaussianMatrix
- Lnorm
- mutualCoherence
- objective1TV
- objectiveL1
- objectiveL2
- oo
- scaleBack.lm
- solve1TV
- solveL1
Description

Utilities for sparse signal recovery suitable for compressed sensing. L1, L2 and TV penalties, DFT basis matrix, simple sparse signal generator, mutual cumulative coherence between two matrices and examples, Lp complex norm, scaling back regression coefficients.

Details

Package: R1magic
Type: Package
Version: 0.3.2
Date: 2014-04-19
License: GPL (>= 3)
LazyLoad: yes

Author(s)

Mehmet Suzen Maintainer: Mehmet Suzen <mehmet.suzen@physics.org>

References

Emmanuel Candes, Justin Romberg, and Terence Tao, Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. (IEEE Trans. on Information Theory, 52(2) pp. 489 - 509, February 2006)


Examples

```
solveL2
sparseSignal
TV1
```

```
compareL1_L2_TV1(100,10,0.1);
```
**CompareL1_L2_TV1**

Compare $L_1$, $L_2$ and TV on a sparse signal.

**Description**

Compare $L_1$, $L_2$ and TV on a sparse signal.

**Usage**

`CompareL1_L2_TV1(N, M, per)`

**Arguments**

- **N**: Size of the sparse signal to generate, integer.
- **M**: Number of measurements.
- **per**: Percentage of spikes.

**Author(s)**

Mehmet Suzen

---

**DFTMatrix0**

Generate Discrete Fourier Transform Matrix using `DFTMatrixPlain`.

**Description**

Generate Discrete Fourier Transform Matrix ($N \times N$).

**Usage**

`DFTMatrix0(N)`

**Arguments**

- **N**: Integer value determines the dimension of the square matrix.

**Value**

It returns a $N \times N$ square matrix.

**Author(s)**

Mehmet Suzen
See Also

DFTMatrixPlain

Examples

DFTMatrixPlain(2)

---

DFTMatrixPlain Generate Plain Discrete Fourier Transform Matrix without the coefficient

Description

Generate plain Discrete Fourier Transform Matrix (NxN) without a coefficient.

Usage

DFTMatrixPlain(N)

Arguments

N Integer value defines the dimension of the square plain DFT matrix.

Value

It returns a NxN square matrix.

Author(s)

Mehmet Suzen

Examples

DFTMatrixPlain(2)
GaussianMatrix

Generate Gaussian Random Matrix (zero mean and standard deviation one.)

Usage

GaussianMatrix(N, M)

Arguments

N Integer value determines number of rows.
M Integer value determines number of columns.

Value

Returns MxN matrix.

Author(s)

Mehmet Suzen

Examples

GaussianMatrix(3, 2)

Lnorm

L-p norm of a given complex vector

Description

L-p norm of a given complex vector

Usage

Lnorm(X, p)

Arguments

X, a complex vector, can be real too.
p, norm value
**Value**

L-p norm of the complex vector

**Author(s)**

Mehmet Suzen

---

**mutualCoherence**

*Cumulative mutual coherence*

**Description**

Generate vector of cumulative mutual coherence of a given matrix up to a given order. Mutual Cumulative Coherence of a Matrix A at order k is defined as

\[ M(A, k) = \max_p \max_{p \neq q, q \in \Omega} \sum_q |\langle a_p, a_q \rangle| / (|a_p| |a_q|) \]

**Usage**

`mutualCoherence(A, k)`

**Arguments**

- **A**
  - A matrix.
- **k**
  - Integer value determines number of columns or the order of mutual coherence function to.

**Value**

Returns k-vector

**Author(s)**

Mehmet Suzen

**References**

Compressed sensing in diffuse optical tomography

**Examples**

```r
set.seed(42)
B <- matrix(rnorm(100), 10, 10) # Gaussian Random Matrix
mutualCoherence(B, 3) # mutual coherence up to order k
```
1-D Total Variation Penalized Objective Function

Description

1-D Total Variation Penalized Objective Function

Usage

objective1TV(x, T, phi, y, lambda)

Arguments

- **x**: Initial value of the vector to be recovered. Sparse representation of the vector (N x 1 matrix) \(X = Tx\), where \(X\) is the original vector
- **T**: Sparsity bases (N x N matrix)
- **phi**: Measurement matrix (M x N).
- **y**: Measurement vector (M x 1).
- **lambda**: Penalty coefficient.

Value

Returns a vector.

Author(s)

Mehmet Suzen

Objective function for ridge L1 penalty

Description

Objective function for ridge L1 penalty

Usage

objectiveL1(x, T, phi, y, lambda)

Arguments

- **x**: unknown vector
- **T**: transform bases
- **phi**: measurement matrix
- **y**: measurement vector
- **lambda**: penalty term
Note

Thank you Jason Xu of Washington University for pointing out complex number handling

Author(s)

Mehmet Suzen

---

**objectiveL2**

*Objective function for Tikhinov L2 penalty*

---

**Description**

Objective function for Tikhinov L2 penalty

**Usage**

`objectiveL2(x, T, phi, y, lambda)`

**Arguments**

- `x`, unknown vector
- `T`, transform bases
- `phi`, measurement matrix
- `y`, measurement vector
- `lambda`, penalty term

**Note**

Thank you Jason Xu of Washington University for pointing out complex number handling

Author(s)

Mehmet Suzen
**Frequency expression for DFT**

**Description**
Frequency expression for DFT

**Usage**

\[ oo(p, \omega) \]

**Arguments**

- \( p \) Exponent
- \( \omega \) Omega expression for DFT

**Author(s)**
Mehmet Suzen

---

**scaleBack.lm**
Transform back multiple regression coefficients to unscaled regression coefficients Original question posed by Mark Seeto on the R mailing list.

**Description**
Transform back multiple regression coefficients to unscaled regression coefficients Original question posed by Mark Seeto on the R mailing list.

**Usage**

\[ \text{scaleBack.lm}(X, Y, \text{betas.scaled}) \]

**Arguments**

- \( X \), unscaled design matrix without the intercept, m by n matrix
- \( Y \), unscaled response, m by 1 matrix
- \( \text{betas.scaled} \), coefficients vector of multiple regression, first term is the intercept

**Note**
2015-04-10
1-D Total Variation Penalized Nonlinear Minimization

Usage

```r
solve1tv(phi, y, x0, lambda = 0.1)
```

Arguments

- `x0`: Initial value of the vector to be recovered. Sparse representation of the vector (N x 1 matrix) X=Tx, where X is the original vector.
- `T`: Sparsity bases (N x N matrix).
- `phi`: Measurement matrix (M x N).
- `y`: Measurement vector (Mx1).
- `lambda`: Penalty coefficient. Defaults 0.1

Value

Returns nlm object.

Author(s)

Mehmet Suzen
solveL1

11 Penalized Nonlinear Minimization

Usage

```
solveL1(phi, y, T, x0, lambda=0.1)
```

Arguments

- `x0`: Initial value of the vector to be recovered. Sparse representation of the vector (N x 1 matrix) X=Tx, where X is the original vector.
- `T`: Sparsity bases (N x N matrix).
- `phi`: Measurement matrix (M x N).
- `y`: Measurement vector (M x 1).
- `lambda`: Penalty coefficient. Defaults 0.1.

Value

Returns nlm object.

Author(s)

Mehmet Suzen

solveL2

12 Penalized Nonlinear Minimization

Usage

```
solveL2(phi, y, T, x0, lambda=0.1)
```

Arguments

- `x0`: Initial value of the vector to be recovered. Sparse representation of the vector (N x 1 matrix) X=Tx, where X is the original vector.
- `T`: Sparsity bases (N x N matrix).
- `phi`: Measurement matrix (M x N).
- `y`: Measurement vector (M x 1).
- `lambda`: Penalty coefficient. Defaults 0.1.
sparsesignal

Description
Sparse digital signal Generator with given thresholds.

Usage
sparsesignal(N, s, b = 1, delta = 1e-07, nlev = 0.05, slev = 0.9)

Arguments
N Number of signal components, vector size.
s Number of spikes, significant components
b Signal bandwidth, defaults 1.
delta Length of discrete distances among components, defaults 1e-7.
nlev Maximum value of insignificant component, relative to b, defaults to 0.05
slev Maximum value of significant component, relative to b, defaults to 0.9

Author(s)
Mehmet Suzen

TV1

Description
1-D total variation of a vector.

Usage
TV1(x)

Arguments
x A vector.
Author(s)

Mehmet Suzen
Index

*Topic package
  R1magic-package, 2

CompareL1_L2_TV1, 3

DFTMatrix0, 3
DFTMatrixPlain, 4

GaussianMatrix, 5

Lnorm, 5

mutualCoherence, 6

objective1TV, 7
objectiveL1, 7
objectiveL2, 8
oo, 9

R1magic (R1magic-package), 2
R1magic-package, 2

scaleBack.1m, 9
solve1TV, 10
solveL1, 11
solveL2, 11
sparseSignal, 12

TV1, 12