Package 'Kernelize'

July 27, 2014

Type Package

Title Computation of Useful Kernel Matrices

Version 1.85

Date 2014-07-28

Author Michael C. Wu, Seunggeun Lee

Maintainer Michael C. Wu <`mcwu@fhcrc.org`>

Description This is simply a useful package for computing a number of kernel matrices in R. The C back end speeds up computation of the n by n kernel matrices.

License GPL(>=2)

LazyLoad yes

Archs i386, x64

R topics documented:

`Kernelize-package` *Package for obtaining kernels. ~~ package title ~~*

Description

This package contains functions for obtaining the kernel matrices for use in the kernel machine test. The back end for some kernels is in C.

Details

Package:	Kernelize
Type:	Package
Version:	1.7
Date:	2012-10-24
License:	GNU General License

Author(s)

Michael C. Wu, Seunggeun Lee

References

None

See Also

None

`getKernel.2wayIX` *Function for estimating the two-way interaction kernel.*

Description

This function estimates the linear kernel based on an arbitrary set of inputs. No missing data are allowed.

Usage

```
getKernel.2wayIX(Z, n, p)
```

Arguments

`Z`	n by p matrix of genetic features (e.g. SNP/SNV/CpG/genes etc)
`n`	Sample size: number of individuals on whom the features have been measured
`p`	The number of measured features.

Value

returns an n by n kernel matrix

Author(s)

Michael C. Wu, Seunggeun Lee

`getKernel.dPoly` *Function for estimating the d-th order polynomial kernel.*

Description

This function estimates the d-th order polynomial kernel based on an arbitrary set of inputs and fixed d. No missing data are allowed.

Usage

```
getKernel.dPoly(Z, n, p, d)
```

Arguments

`Z`	n by p matrix of genetic features (e.g. SNP/SNV/CpG/genes etc)
`n`	Sample size: number of individuals on whom the features have been measured
`p`	The number of measured features.
`d`	The order of the polynomial.

Value

returns an n by n kernel matrix

Author(s)

Michael C. Wu, Seunggeun Lee

`getKernel.gaussian` *Function for estimating the gaussian kernel.*

Description

This function estimates the gaussian kernel based on an arbitrary set of inputs. No missing data are allowed.

Usage

```
getKernel.gaussian(Z, n, p, rho = 1)
```

Arguments

`Z`	n by p matrix of genetic features (e.g. SNP/SNV/CpG/genes etc)
`n`	Sample size: number of individuals on whom the features have been measured
`p`	The number of measured features.
`rho`	The scale parameter for the gaussian kernel.

Value

returns an n by n kernel matrix

Author(s)

Michael C. Wu, Seunggeun Lee

`getKernel.IBS` *Function for estimating the IBS kernel.*

Description

This function estimates the IBS kernel based on an arbitrary set of inputs. No missing data are allowed.

Usage

```
getKernel.IBS(Z, n, p)
```

Arguments

`Z`	n by p matrix of genetic features (e.g. SNP/SNV etc). This function only really works for genotype data.
`n`	Sample size: number of individuals on whom the features have been measured
`p`	The number of measured features.

Value

returns an n by n kernel matrix

Author(s)

Michael C. Wu, Seunggeun Lee

`getKernel.IBS.weighted` *Function for estimating the weighted IBS kernel.*

Description

This function estimates the weighted IBS kernel based on an arbitrary set of inputs. No missing data are allowed. By default the weights are set to be 1/MAF.

Usage

```
getKernel.IBS.weighted(Z, n, p, weights)
```

Arguments

`Z`	n by p matrix of genetic features (e.g. SNP/SNV/CpG/genes etc)
`n`	Sample size: number of individuals on whom the features have been measured
`p`	The number of measured features.
`weights`	The weights for the individual variants in the study. They default to 1/MAF

Value

returns an n by n kernel matrix

Author(s)

Michael C. Wu, Seunggeun Lee

`getKernel.linear` *Function for estimating the linear kernel.*

Description

This function estimates the linear kernel based on an arbitrary set of inputs. No missing data are allowed.

Usage

```
getKernel.linear(Z, n, p)
```

Arguments

`Z`	n by p matrix of genetic features (e.g. SNP/SNV/CpG/genes etc)
`n`	Sample size: number of individuals on whom the features have been measured
`p`	The number of measured features.

Value

returns an n by n kernel matrix

Author(s)

Michael C. Wu, Seunggeun Lee

`getKernel.quadratic` *Function for estimating the quadratic kernel.*

Description

This function estimates the quadratic kernel based on an arbitrary set of inputs. No missing data are allowed.

Usage

```
getKernel.quadratic(Z, n, p)
```

Arguments

`Z`	n by p matrix of genetic features (e.g. SNP/SNV/CpG/genes etc)
`n`	Sample size: number of individuals on whom the features have been measured
`p`	The number of measured features.

Value

returns an n by n kernel matrix

Author(s)

Michael C. Wu, Seunggeun Lee

`getKernel.squareDiff` *Function for estimating the Square Difference kernel.*

Description

This function estimates the Square Difference kernel based on an arbitrary set of inputs. No missing data are allowed.

Usage

```
getKernel.squareDiff(Z, n, p)
```

Arguments

`Z`	n by p matrix of genetic features (e.g. SNP/SNV/CpG/genes etc)
`n`	Sample size: number of individuals on whom the features have been measured
`p`	The number of measured features.

Value

returns an n by n kernel matrix

Author(s)

Michael C. Wu, Seunggeun Lee

`getKernel.triangle` *Function for estimating the triangle kernel.*

Description

This function estimates the triangle kernel based on an arbitrary set of inputs. No missing data are allowed.

Usage

```
getKernel.triangle(Z, n, p)
```

Arguments

`Z`	n by p matrix of genetic features (e.g. SNP/SNV/CpG/genes etc)
`n`	Sample size: number of individuals on whom the features have been measured
`p`	The number of measured features.

Value

returns an n by n kernel matrix

Author(s)

Michael C. Wu, Seunggeun Lee

Index