

# Package ‘VSE’

March 21, 2016

**Type** Package

**Title** Variant Set Enrichment

**Version** 0.99

**Date** 2016-03-03

**Author** Musaddeque Ahmed <musa.ahmed@utoronto.ca>

**Maintainer** Hansen He <hansenhe@uhnresearch.ca>

**Description** Calculates the enrichment of associated variant set (AVS) for an array of genomic regions. The AVS is the collection of disjoint LD blocks computed from a list of disease associated SNPs and their linked (LD) SNPs. VSE generates a null distribution of matched random variant sets (MRVSS) from 1000 Genome Project Phase III data that are identical to AVS, LD block by block. It then computes the enrichment of AVS intersecting with user provided genomic features (e.g., histone marks or transcription factor binding sites) compared with the null distribution.

**License** GPL-3

**LazyData** TRUE

**RoxygenNote** 5.0.1.9000

**Depends** GenomicRanges

**Suggests** knitr, rmarkdown, ggplot2

**Imports** igraph, car, IRanges, parallel

**VignetteBuilder** knitr

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2016-03-21 23:10:53

## R topics documented:

|                           |   |
|---------------------------|---|
| avsSize . . . . .         | 2 |
| bedToGRanges . . . . .    | 2 |
| intersectMatrix . . . . . | 3 |
| loadLd . . . . .          | 4 |

|                                |   |
|--------------------------------|---|
| loadSampleRegions . . . . .    | 4 |
| makeAVS . . . . .              | 5 |
| makeMRVS . . . . .             | 5 |
| variantSetEnrichment . . . . . | 6 |
| VSEplot . . . . .              | 7 |
| VSEqq . . . . .                | 7 |
| VSESummary . . . . .           | 8 |

|              |          |
|--------------|----------|
| <b>Index</b> | <b>9</b> |
|--------------|----------|

---

|         |                |
|---------|----------------|
| avsSize | <i>avsSize</i> |
|---------|----------------|

---

### Description

This function will output a dataframe of LD size of each tag SNP

### Usage

```
avsSize(avs)
```

### Arguments

avs                    A GRanges object which is outputted by loadLd function

### Examples

```
ld<-loadLd(file.path(system.file("extdata", "ld_BCa_raggr.csv", package="VSE")), type="raggr")
avs<-makeAVS(ld)
avsSize(avs)
```

---

|              |                     |
|--------------|---------------------|
| bedToGRanges | <i>bedToGRanges</i> |
|--------------|---------------------|

---

### Description

This function will convert a bed to GRanges object

### Usage

```
bedToGRanges(file)
```

### Arguments

file                    A bed file. Must contain at least three columns: chr, start and end.

**Examples**

```
## Not run:  
bedToGRanges(file)  
  
## End(Not run)
```

---

|                 |                        |
|-----------------|------------------------|
| intersectMatrix | <i>intersectMatrix</i> |
|-----------------|------------------------|

---

**Description**

This function will count the intersection tally of AVS and genomic regions

**Usage**

```
intersectMatrix(avs, regions, ...)
```

**Arguments**

|         |  |
|---------|--|
| avs     | A GRanges object which is outputted by loadLd function   |
| regions | A data frame. The data frame contains sample sheet identical to DiffBind or ChIPQC input sample sheets |
| ...     | Heatmap arguments  |

**Examples**

```
## Not run:  
intersectMatrix(avs,  
               regions=samples,  
               col=c("white","grey10"),  
               scale="none",  
               margins=c(10,5),  
               cexRow = 1,  
               cexCol = 0.5,  
               Rowv=NA,  
               Colv=NA)  
  
## End(Not run)
```

---

|        |               |
|--------|---------------|
| loadLd | <i>loadLd</i> |
|--------|---------------|

---

### Description

This function will load a raggr output csv or user defined bed file to GRanges object. It is recommended to generate a LD file using <http://raggr.usc.edu>. If you prefer to use other softwares to calculate LD SNPs (e.g., plink), please format the output to bed files. Please make sure that you use a cutoff of 0.8 for r2 value.

### Usage

```
loadLd(data, type)
```

### Arguments

|      |  |
|------|--|
| data | An input file. Must be a raggr generated csv file or a bed file. If a bed file, it must contain at least five columns: chr, start, end, LD_snp_id, tag_snp_id. |
| type | The type of input file. Must be "bed" or "raggr".  |

### Examples

```
ld<-loadLd(file.path(system.file("extdata", "ld_BCa_raggr.csv", package="VSE")), type="raggr")
```

---

|                   |                          |
|-------------------|--------------------------|
| loadSampleRegions | <i>loadSampleRegions</i> |
|-------------------|--------------------------|

---

### Description

The sample bed files are DNase-seq, ChIP-seq for H3K4me1, H3K4me3, H3K27ac, H3K27me3 and H3K36me3 for MCF7 cells. The data are obtained from ENCODE project. There is also one sampleSheet.csv which is the sample sheet for the bed regions in the format similar to ChIPQC or DiffBind requirement.

### Usage

```
loadSampleRegions()
```

### Details

This function will download sample bed files from [www.hanshelab.org/VSE/sample\\_regions](http://www.hanshelab.org/VSE/sample_regions) in /VSE\_samples

### Value

A directory names VSE\_samples that will contain 6 bed files and 1 sampleSheet.csv

**Examples**

```
## Not run:
loadSampleRegions()

## End(Not run)
```

---

|         |                |
|---------|----------------|
| makeAVS | <i>makeAVS</i> |
|---------|----------------|

---

**Description**

This function will create disjoint LD blocks from a GRanges object

**Usage**

```
makeAVS(x)
```

**Arguments**

x                    A GRanges object which is outputted by loadLd function

**Examples**

```
ld<-loadLd(file.path(system.file("extdata", "ld_BCa_raggr.csv", package="VSE")), type="raggr")
avs<-makeAVS(ld)
```

---

|          |                 |
|----------|-----------------|
| makeMRVS | <i>makeMRVS</i> |
|----------|-----------------|

---

**Description**

This function will calculate matching random variant sets (MRVS) identical to AVS

**Usage**

```
makeMRVS(avs, bgSize = 100, mc.cores = 6)
```

**Arguments**

avs                    A GRanges object which is outputted by makeAVS function  
bgSize                 An integer for the number of MRVS. Default: 100  
mc.cores               Number of cores to use. Default: 8

## Examples

```
## Not run:
ld <- loadLd("ld.csv", type="raggr")
avs <- makeAVS(ld)
makeMRVS(avs, bgSize=100, mc.cores=8)

## End(Not run)
#As an example, we have added MRVS (size=200) for Breast Cancer AVS.
load(file.path(system.file("extdata", "bca.mrvs.200.Rda", package="VSE")))
```

---

variantSetEnrichment *variantSetEnrichment*

---

## Description

This function will calculate matching random variant sets (MRVS) identical to AVS

## Usage

```
variantSetEnrichment(avs, mrvs, regions)
```

## Arguments

|         |  |
|---------|--|
| avs     | A GRangesList object which is outputted by makeAVS function  |
| mrvs    | A list of GRangesList objects outputted by makeMRVS function   |
| regions | A data frame. The data frame contains sample sheet identical to DiffBind or ChIPQC input sample sheets |

## Examples

```
## Not run:
variantSetEnrichment(avs, mrvs, regions=samples)
#We have included the output from our example analysis with the package. \
You can load the example VariantSetEnrichment output by typing:
load(file.path(system.file("extdata", "vse_output.Rda", package="VSE")))
```

## End(Not run)

---

VSEplot

*VSEplot*

---

### Description

This function will generate a figure for VSE analysis

### Usage

```
VSEplot(data, padj = 0.01, ...)
```

### Arguments

|                   |  |
|-------------------|--|
| <code>data</code> | A list of matrices outputted by the function <code>VariantSetEnrichment</code> |
| <code>padj</code> | Bonferroni adjusted p-value cutoff. Default: 0.01                              |
| <code>...</code>  | Arguments from <code>boxplot</code>  |

### Examples

```
#Load pre-saved object "bca.vse" as an example VSE output
load(file.path(system.file("extdata", "vse_output.Rda", package="VSE")))
VSEplot(bca.vse, las=2, pch=20, cex=1, cex.main=0.6, padj=0.05)
```

---

VSEqq

*VSEqq*

---

### Description

This function will generate QQ plots of normal distribution

### Usage

```
VSEqq(data, ...)
```

### Arguments

|                   |  |
|-------------------|--|
| <code>data</code> | A matrix outputted by the function <code>VariantSetEnrichment</code> |
| <code>...</code>  | Arguments for <code>qqnorm</code> plot                               |

### Examples

```
#Load pre-saved object "bca.vse" as an example VSE output
load(file.path(system.file("extdata", "vse_output.Rda", package="VSE")))
VSEqq(bca.vse)
```

VSESummary

*VSESummary*

---

**Description**

This function will compute the enrichment from a VSE matrix

**Usage**

```
VSESummary(data)
```

**Arguments**

`data`            A matrix outputted by the function VariantSetEnrichment

**Examples**

```
#Load pre-saved object "bca.vse" as an example VSE output
load(file.path(system.file("extdata", "vse_output.Rda", package="VSE")))
VSESummary(bca.vse)
```



# Index

- \*Topic **AVS,Granges**
    - avsSize, 2
    - intersectMatrix, 3
    - makeAVS, 5
  - \*Topic **LD,bed,raggr,GRanges**
    - loadLd, 4
  - \*Topic **VSE,**
    - VSEplot, 7
    - VSEqq, 7
    - VSESummary, 8
  - \*Topic **VSE**
    - makeMRVS, 5
    - variantSetEnrichment, 6
  - \*Topic **bed,Granges**
    - bedToGRanges, 2
  - \*Topic **boxplot**
    - VSEplot, 7
    - VSEqq, 7
  - \*Topic **enrichment**
    - VSESummary, 8
  - \*Topic **sample,histone,MCF7**
    - loadSampleRegions, 4
- avsSize, 2
- bedToGRanges, 2
- intersectMatrix, 3
- loadLd, 4
- loadSampleRegions, 4
- makeAVS, 5
- makeMRVS, 5
- variantSetEnrichment, 6
- VSEplot, 7
- VSEqq, 7
- VSESummary, 8