

# Package ‘RareVariantVis’

June 5, 2026

**Type** Package

**Title** A suite for analysis of rare genomic variants in whole genome sequencing data

**Version** 2.41.0

**Date** 2018-04-15

**Author** Adam Gudys and Tomasz Stokowy

**Maintainer** Tomasz Stokowy <tomasz.stokowy@k2.uib.no>

**Description** Second version of RareVariantVis package aims to provide comprehensive information about rare variants for your genome data.

It annotates, filters and presents genomic variants (especially rare ones) in a global, per chromosome way. For discovered rare variants CRISPR guide RNAs are designed, so the user can plan further functional studies. Large structural variants, including copy number variants are also supported.

Package accepts variants directly from variant caller - for example GATK or Speedseq. Output of package are lists of variants together with adequate visualization.

Visualization of variants is performed in two ways -

standard that outputs png figures and interactive that uses

JavaScript d3 package. Interactive visualization allows to analyze trio/family data, for example in search for causative

variants in rare Mendelian diseases, in point-and-click interface. The package includes homozygous region caller and allows to analyse whole human genomes in less than 30 minutes on a desktop computer.

RareVariantVis disclosed novel causes of several rare monogenic disorders, including one with non-coding causative variant - keratolythic winter erythema.

**License** Artistic-2.0

**LazyData** TRUE

**Depends** BiocGenerics, VariantAnnotation, googleVis, GenomicFeatures

**Imports** S4Vectors, IRanges, GenomeInfoDb, GenomicRanges, gtools, BSgenome, BSgenome.Hsapiens.UCSC.hg19, TxDb.Hsapiens.UCSC.hg19.knownGene, phastCons100way.UCSC.hg19, SummarizedExperiment, GenomicScores

**Suggests** knitr

**VignetteBuilder** knitr

**biocViews** GenomicVariation, Sequencing, WholeGenome

**NeedsCompilation** no

**git\_url** <https://git.bioconductor.org/packages/RareVariantVis>

**git\_branch** devel  
**git\_last\_commit** 36b9f76  
**git\_last\_commit\_date** 2026-04-28  
**Repository** Bioconductor 3.24  
**Date/Publication** 2026-06-04

## Contents

callHomozygous . . . . .	2
chromosomeVis . . . . .	3
getCrisprGuides . . . . .	4
movingAverage . . . . .	5
multipleVis . . . . .	6
rareVariantVis . . . . .	7
<b>Index</b>	<b>8</b>

---

callHomozygous	<i>Call homozygous regions from sequencing data</i>
----------------	---

---

## Description

Function calls homozygous regions from whole genome sequencing data.

## Usage

```
callHomozygous(sample, chromosomes, caller = "speedseq", MA_Window = 1000, HMZ_length = 100000, min_n_HMZ = 20)
```

## Arguments

sample	A name of SNV sample file to be analyzed.
chromosomes	A vector of strings indicating chromosomes to be analyzed.
caller	A string indicating vcf caller. Default is "speedseq", supports "GATK"
MA_Window	A number indicating window size for moving average function. Recommended value for genome is 2000, for exome is 20. Default is 1000.
HMZ_length	Minimal length of homozygous region to be called. Default is 100000.
min_n_HMZ	Minimal number of variants necessary to call a region. Default is 20.

## Value

comp1	function calls homozygous regions from whole genome sequencing data and returns them in a tab separated txt file.
-------	---

## Author(s)

Tomasz Stokowy

**Examples**

```
# sample = system.file("extdata", "CoriellIndex_S1_chr19_9-10_S1.vcf.recode.vcf.gz",
#                       package = "RareVariantVis")
# callHomozygous(sample=sample, chromosomes=c("19"))
```

---

chromosomeVis

*Visualization of all genomic variants on the chromosome*


---

**Description**

Reads files containing single nucleotide variants (SNV) and structural genomic variants (SV) - vcf.gz files generated by speedseq aligner and variant caller. Function outputs visualization png figures. Figure illustrates variants (blue dots) in their genomic coordinates (x axis). Ratio of alternative reads and depth (y axis) gives information about type of variant: homozygous alternative (expected ratio 1) and heterozygous (expected ratio 0.5). Green dots represent rare variants that pass filters: coding/UTR, nonsynonymous variant with dbSNP frequency < 0.01 and ExAC frequency < 0.01. Orange vertical lines depict position of centromere. Orange dots depict structural and copy number variants that overlap with coding region and are relatively good quality (QUAL > 0). Red curve illustrates moving average of alternative reads/depth ratio. High values of this curve (exceeding 0.75) can suggest potential homozygous/deleterious regions. In addition, files containing table with rare SNV and SV variants only are generated. Tables include variants that passed filters specified above with annotations (uniprot, RefSeq and other). Function analyzes whole genome in about 30 minutes on a desktop computer.

**Usage**

```
chromosomeVis(sample, sv_sample, dbSNP_file, Exac_file, chromosomes, pngWidth, pngHeight, caller, M
```

**Arguments**

sample	A name of SNV sample file to be analyzed.
sv_sample	A name of additional SV sample file. If not specified, structural variants are discarded.
dbSNP_file	A file with SNPs database. If not specified, chromosome 19 dbSNP is used.
Exac_file	ExAC database file. If not specified, chromosome 19 ExAC is used.
chromosomes	A vector of strings indicating chromosomes to be analyzed.
pngWidth	A number indicating pixel width of output png files. Default is 1600.
pngHeight	A number indicating pixel height of output png files. Default is 1200.
caller	A string indicating vcf caller. Default is "speedseq", supports "GATK"
MA_Window	A number indicating window size for moving average function. Recommended value for genome is 2000, for exome is 20. Default is 1000.
coding_regions_file	A bed file indicating coding regions
annotation_file	Text file indicating positions of the genes (from UCSC)
uniprot_file	Text file indicating gene functions and related diseases (from Uniprot)

**Value**

comp1                    function plots static visualization of genomic variants on all chromosomes, annotates them, filters and reports output variants in tables

**Author(s)**

Adam Gudys and Tomasz Stokowy

**Examples**

```
# analyze chromosome 19 from example genome
sample = system.file("extdata", "CoriellIndex_S1_chr19_9-10_S1.vcf.recode.vcf.gz",
  package = "RareVariantVis")
sv_sample = system.file("extdata", "CoriellIndex_S1.sv.vcf.gz",
  package = "RareVariantVis")
chromosomeVis(sample=sample, sv_sample=sv_sample, chromosomes=c("19"))

# without sv data
# sample = system.file("extdata", "CoriellIndex_S1_chr19_9-10_S1.vcf.recode.vcf.gz",
#   package = "RareVariantVis")
# chromosomeVis(sample=sample, chromosomes=c("19"))

# analyze entire genome (use external full-genome dbSNP and ExAC)
# it takes approximately 30 mins on a desktop computer
# large example data and all necessary hg19 references can be downloaded from:
# https://github.com/agudys/DataRareVariantVis
# dbSNP_file = "All_20160601.vcf.gz"
# Exac_file = "ExAC.r0.3.1.sites.vep.vcf.gz"
# chromosomeVis(sample=sample, sv_sample=sv_sample,
#   dbSNP_file=dbSNP_file, Exac_file=Exac_file,
#   chromosomes=c(as.character(1:22), "X", "Y"), MA_Window = 2000,
#   coding_regions_file = "nexterarapidcapture_exome_targetedregions_v1.2.bed",
#   annotation_file = "UCSC_hg19_refSeq_160702.txt",
#   uniprot_file = "uniprot-all.txt")
```

---

getCrisprGuides

*Retrieve CRISPR/Cas9 guides.*

---

**Description**

Function checks whether a guideRNA can be found that overlaps given SNP. Returns sequence of the guideRNA with the variant marked with the lowercase letters. When multiple guideRNAs are possible for given SNP, guideRNA with the variant closest to the PAM site is being selected.

**Arguments**

df                    A data frame, preferably out from chromosomeVis.  
genome                A object of the BSGenome, by default BSGenome.Hsapiens.UCSC.hg19.  
gsize                 Preferred size of the guideRNA, by default, standard 23 is used.

PAM	Preferred Protospacer Adjacent Motif "PAM", short motif that has to be found on the 5' end of the guideRNA, by default, Cas9 is used "GG".
PAM_rev	Preferred Protospacer Adjacent Motif "PAM", short motif that has to be found on the reverse strand, by default, Cas9 is used "CC". This is checked only when no guideRNA is found on the forward strand.

**Value**

character vector

Vector of guideRNAs, when no guideRNA was found for the forward strand, reverse strand is checked, when no guideRNA is found NA is returned.

**Author(s)**

Kornel Labun

**Examples**

```
file <- system.file("extdata", "RareVariants_CoriellIndex_S1.txt",
  package = "RareVariantVis")
df <- read.delim(file, stringsAsFactors = FALSE)
getCrisprGuides(df)
```

---

movingAverage

*Computation of moving average*

---

**Description**

Function calculates moving average from a vector of numeric values.

**Usage**

```
movingAverage(x, n, centered)
```

**Arguments**

x	a vector of numeric values for which moving average is computed
n	numeric value giving the frame length for moving average
centered	logic variable indicating if moving average should be centered (default = FALSE)

**Value**

comp1 function returns vector of moving average values

**Author(s)**

Winston Chang

**Examples**

```
movingAverage(1:20, n=3, centered=FALSE)
```

---

multipleVis	<i>Interactive visualization of rare variants on the chromosome, applicable for multiple files</i>
-------------	--

---

## Description

Reads files containing table of rare variants from one chromosome and provides adequate multiple sample visualization. Input files can be obtained from function `chromosomeVis`. Function outputs visualization html figure. Figure depicts samples in subfigures. Subfigures illustrate variants (dots) in their genomic coordinates (x axis). Ratio of alternative reads and depth (y axis) gives information about type of variant: homozygous alternative (expected ratio 1) and heterozygous (expected ratio 0.5). Zoom to the figures is possible, by marking the region of interest with mouse left click. Right click induces zoom out and return to the original plot. Pointing on variants provides basic information about the variant - gene name and position on chromosome.

## Usage

```
multipleVis(inputFiles, outputFile, sampleNames, chromosome)
```

## Arguments

<code>inputFiles</code>	Vector of strings containing input file names.
<code>outputFile</code>	Output file name (string).
<code>sampleNames</code>	Vector of sample names (strings).
<code>chromosome</code>	Name of the chromosome to be analyzed (string).

## Value

<code>comp1</code>	function returns html visualization file for specified samples
--------------------	--

## Author(s)

Adam Gudys and Tomasz Stokowy

## Examples

```
file1 = system.file("extdata", "RareVariants_CoriellIndex_S1.txt",  
                   package = "RareVariantVis")  
file2 = system.file("extdata", "RareVariants_Coriell_S2.txt",  
                   package = "RareVariantVis")  
inputFiles = c(file1, file2)  
sampleNames = c("CoriellIndex_S1", "Coriell_S2");  
multipleVis(inputFiles, "CoriellSamples.html", sampleNames, "19")
```

---

`rareVariantVis`*Interactive visualization of rare variants on the chromosome*

---

### Description

Reads file containing table of rare variants from one chromosome and provides adequate visualization. Input file can be obtained from function `chromosomeVis`. Function outputs visualization html figure in current working directory. Figure illustrates variants (dots) in their genomic coordinates (x axis). Ratio of alternative reads and depth (y axis) gives information about type of variant: homozygous alternative (expected ratio 1) and heterozygous (expected ratio 0.5). Zoom to the figures is possible, by marking the region of interest with mouse left click. Right click induces zoom out and return to the original plot. Pointing on variants provides basic information about the variant - gene name and position on chromosome.

### Usage

```
rareVariantVis(input, outputFile, sample, chromosomes, append)
```

### Arguments

<code>input</code>	Name of the input file (string) containing the table with rare variants generated by <code>chromosomeVis</code> . It can also be the variable with the table itself.
<code>outputFile</code>	Name of the output file (string) with visualisation.
<code>sample</code>	Name of the sample (used only for entitling the charts).
<code>chromosomes</code>	Vector of chromosome names (strings) to be included in the visualisation (all chromosomes by default). Chromosomes included in the parameter but absent in the table are omitted.
<code>append</code>	Logical value indicating whether charts should be appended to the output file without destroying it (FALSE by default).

### Value

`comp1` function returns html file with visualization of rare variants

### Author(s)

Adam Gudys and Tomasz Stokowy

### Examples

```
file = system.file("extdata", "RareVariants_CoriellIndex_S1.txt",  
  package = "RareVariantVis")  
rareVariantVis(input=file, "RareVariants_CoriellIndex_S1.html", "CorielIndex")
```

# Index

- \* **~CRISPR/Cas9**
    - getCrisprGuides, 4
  - \* **~call homozygous regions**
    - callHomozygous, 2
  - \* **~moving average**
    - movingAverage, 5
  - \* **~rare variants**
    - multipleVis, 6
    - rareVariantVis, 7
  - \* **~variants**
    - chromosomeVis, 3
  - \* **~visualization**
    - chromosomeVis, 3
    - multipleVis, 6
    - rareVariantVis, 7
- callHomozygous, 2
- chromosomeVis, 3
- getCrisprGuides, 4
- movingAverage, 5
- multipleVis, 6
- rareVariantVis, 7