

# Package ‘rfPred’

November 13, 2024

**Type** Package

**Title** Assign rfPred functional prediction scores to a missense variants list

**Version** 1.44.0

**Date** 2021-11-21

**Author** Fabienne Jabot-Hanin, Hugo Varet and Jean-Philippe Jais

**Depends** R (>= 3.5.0), methods

**Imports** utils, GenomeInfoDb, data.table, IRanges, GenomicRanges, parallel, Rsamtools

**Suggests** BiocStyle

**Maintainer** Hugo Varet <varethugo@gmail.com>

**Description** Based on external numerous data files where rfPred scores are pre-calculated on all genomic positions of the human exome, the package gives rfPred scores to missense variants identified by the chromosome, the position (hg19 version), the referent and alternative nucleotids and the uniprot identifier of the protein. Note that for using the package, the user has to download the TabixFile and index (approximately 3.3 Go).

**License** GPL (>=2 )

**Encoding** UTF-8

**biocViews** Software, Annotation, Classification

**RoxygenNote** 7.2.1

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

**git\_branch** RELEASE\_3\_20

**git\_last\_commit** 9a5122d

**git\_last\_commit\_date** 2024-10-29

**Repository** Bioconductor 3.20

**Date/Publication** 2024-11-12

## Contents

rfPred-package . . . . .	2
example_GRanges . . . . .	2

rfPred_scores . . . . .	3
rfPred_scores_motor . . . . .	6
variant_list_Y . . . . .	7

<b>Index</b>	<b>8</b>
--------------	----------

---

rfPred-package	<i>Assign functional prediction rfPred scores to human missense variants (random forest method based on SIFT, Polyphen2, PhyloP, LRT and Mutation Taster)</i>
----------------	---

---

### Description

The package provides a function which returns the rfPred score for a list of non-synonymous missense variants. All the rfPred scores are pre-calculated and stored in a TabixFile available on a server and which can be downloaded for using the package while not connected on the Internet. The package does not work without an access to the TabixFile. However, a toy example on the chromosome Y is available within the package to test the rfPred\_scores function. curves with numbers of subjects at risk, compare data sets, display spaghetti-plot, build multi-contingency tables...

### Author(s)

Fabienne Jabot-Hanin, Hugo Varet and Jean-Philippe Jais

### References

dbNSFP database: Liu X, Jian X and Boerwinkle E. 2011. dbNSFP: a lightweight database of human non-synonymous SNPs and their functional predictions. *Human Mutation*. 32:894-899.

rfPred method: Jabot-Hanin F, Varet H, Tores F and Jais J-P. 2013. rfPred: a new meta-score for functional prediction of missense variants in human exome (submitted).

---

example_GRanges	<i>Toy example of GRanges object</i>
-----------------	--------------------------------------

---

### Description

Toy example of GRanges object

### Format

A GRanges object with 11 rows and several columns:

seqnames Chromosome number (only Y in this example)

ranges IRanges object for which start=end: position on the chromosome

reference Referent nucleotid (A, C, G or T)

alteration Alteration nucleotid (A, C, G or T)

---

rfPred_scores	<i>Assign functional prediction rfPred scores to human missense variants</i>
---------------	--

---

### Description

rfPred is a statistical method which combines 5 algorithms predictions in a random forest model: SIFT, Polyphen2, LRT, PhyloP and MutationTaster. These scores are available in the dbNFSP database for all the possible missense variants in hg19 version, and the package rfPred gives a composite score more reliable than each of the isolated algorithms.

### Usage

```
rfPred_scores(
  variant_list,
  data = system.file("extdata/chrY_rfPred.txtz", package = "rfPred"),
  index = system.file("extdata/chrY_rfPred.txtz.tbi", package = "rfPred"),
  all.col = FALSE,
  file.export = NULL,
  n.cores = 1
)

rfPred_scores(variant_list,
              data=system.file("extdata/chrY_rfPred.txtz", package="rfPred"),
              index=system.file("extdata/chrY_rfPred.txtz.tbi", package="rfPred"),
              all.col=FALSE, file.export=NULL, n.cores=1)

## S4 method for signature 'character'
rfPred_scores(
  variant_list,
  data = system.file("extdata/chrY_rfPred.txtz", package = "rfPred"),
  index = system.file("extdata/chrY_rfPred.txtz.tbi", package = "rfPred"),
  all.col = FALSE,
  file.export = NULL,
  n.cores = 1
)

## S4 method for signature 'GRanges'
rfPred_scores(
  variant_list,
  data = system.file("extdata/chrY_rfPred.txtz", package = "rfPred"),
  index = system.file("extdata/chrY_rfPred.txtz.tbi", package = "rfPred"),
  all.col = FALSE,
  file.export = NULL,
  n.cores = 1
)
```

### Arguments

**variant\_list** A variants list in a data.frame containing 4 or 5 columns: chromosome number, hg19 genomic position on the chromosome, reference nucleotide, variant nucleotide and uniprot protein identifier (optional); or a character string of the

	path to a VCF (Variant Call Format) file; or a GRanges object with metadata containing textually reference, alteration and proteine (optional) columns names for reference and alteration
data	Path to the compressed TabixFile, either on the server (default) or on the user's computer
index	Path to the index of the TabixFile, either on the server (default) or on the user's computer
all.col	TRUE to return all available information, FALSE to return a more compact result (the most informative columns, see Value)
file.export	Optional, name of the CSV file in which export the results (default is NULL)
n.cores	number of cores to use when scanning the TabixFile, can be efficient for large request (default is 1)

### Value

The variants list with the assigned rfPred scores, as well as the scores used to build rfPred meta-score: SIFT, phyloP, MutationTaster, LRT (transformed) and Polyphen2 (corresponding to Polyphen2\_HVAR\_score). The data frame returned contains these columns:

chromosome	chromosome number
position_hg19	physical position on the chromosome as to hg19 (1-based coordinate)
reference	reference nucleotide allele (as on the + strand)
alteration	alternative nucleotide allele (as on the + strand)
proteine	Uniprot accession number
aaref	reference amino acid
aaalt	alternative amino acid
aapos	amino acid position as to the protein
rfPred_score	rfPred score between 0 and 1 (higher it is, higher is the probability of pathogenicity)
SIFT_score	SIFT score between 0 and 1 (higher it is, higher is the probability of pathogenicity contrary to the original SIFT score) = 1-original SIFT score
Polyphen2_score	Polyphen2 (HVAR one) score between 0 and 1, used to calculate rfPred (higher it is, higher is the probability of pathogenicity)
MutationTaster_score	MutationTaster score between 0 and 1 (higher it is, higher is the probability of pathogenicity)
PhyloP_score	PhyloP score between 0 and 1 (higher it is, higher is the probability of pathogenicity): $\text{PhyloP\_score} = 1 - 0.5 \times 10^{\text{phyloP}}$ if $\text{phyloP} > 0$ or $\text{PhyloP\_score} = 0.5 \times 10^{-\text{phyloP}}$ if $\text{phyloP} < 0$
LRT_score	LRT score between 0 and 1 (higher it is, higher is the probability of pathogenicity): $\text{LRT\_score} = 1 - \text{LRT\_original} \times 0.5$ if $\text{LRT\_Omega} < 1$ or $\text{LRT\_score} = \text{LRT\_original} \times 0.5$ if $\text{LRT\_Omega} \geq 1$

The following columns are also returned if all.col is TRUE:

Uniprot_id	Uniprot ID number
genename	gene name

position_hg18	physical position on the chromosome as to hg18 (1-based coordinate)
Polyphen2_HDIV_score	Polyphen2 score based on HumDiv, i.e. hdiv_prob. The score ranges from 0 to 1: the corresponding prediction is "probably damaging" if it is in [0.957,1]; "possibly damaging" if it is in [0.453,0.956]; "benign" if it is in [0,0.452]. Score cut-off for binary classification is 0.5, i.e. the prediction is "neutral" if the score is lower than 0.5 and "deleterious" if the score is higher than 0.5. Multiple entries separated by ";"
Polyphen2_HDIV_pred	Polyphen2 prediction based on HumDiv: D (probably damaging), P (possibly damaging) and B (benign). Multiple entries separated by ";"
Polyphen2_HVAR_score	Polyphen2 score based on HumVar, i.e. hvar_prob. The score ranges from 0 to 1, and the corresponding prediction is "probably damaging" if it is in [0.909,1]; "possibly damaging" if it is in [0.447,0.908]; "benign" if it is in [0,0.446]. Score cut-off for binary classification is 0.5, i.e. the prediction is "neutral" if the score is lower than 0.5 and "deleterious" if the score is higher than 0.5. Multiple entries separated by ";"
Polyphen2_HVAR_pred	Polyphen2 prediction based on HumVar: D (probably damaging), P (possibly damaging) and B (benign). Multiple entries separated by ";"
MutationTaster_pred	MutationTaster prediction: A (disease_causing_automatic), D (disease_causing), N (polymorphism) or P (polymorphism_automatic)
phyloP	original phyloP score
LRT_Omega	estimated nonsynonymous-to-synonymous-rate ratio
LRT_pred	LRT prediction, D(eleterious), N(eutral) or U(nknown)

**Author(s)**

Fabienne Jabot-Hanin, Hugo Varet and Jean-Philippe Jais

**References**

Jabot-Hanin F, Varet H, Tores F and Jais J-P. 2013. rfPred: a new meta-score for functional prediction of missense variants in human exome (submitted).

**Examples**

```
# from a data.frame without uniprot protein identifier
data(variant_list_Y)
res=rfPred_scores(variant_list = variant_list_Y[,1:4],
                  data = system.file("extdata", "chrY_rfPred.txtz", package="rfPred",mustWork=TRUE),
                  index = system.file("extdata", "chrY_rfPred.txtz.tbi", package="rfPred",mustWork=TRUE))
# from a data.frame with uniprot protein identifier
res2=rfPred_scores(variant_list = variant_list_Y,
                  data = system.file("extdata", "chrY_rfPred.txtz", package="rfPred",mustWork=TRUE),
                  index = system.file("extdata", "chrY_rfPred.txtz.tbi", package="rfPred",mustWork=TRUE))
# from a VCF file
res3=rfPred_scores(variant_list = system.file("extdata", "example.vcf", package="rfPred",mustWork=TRUE),
                  data = system.file("extdata", "chrY_rfPred.txtz", package="rfPred",mustWork=TRUE),
                  index = system.file("extdata", "chrY_rfPred.txtz.tbi", package="rfPred",mustWork=TRUE))
```

```
# from a GRanges object
data(example_GRanges)
res4=rfPred_scores(variant_list = example_GRanges,
                   data = system.file("extdata", "chrY_rfPred.txtz", package="rfPred",mustWork=TRUE),
                   index = system.file("extdata", "chrY_rfPred.txtz.tbi", package="rfPred",mustWork=TRUE))
```

---

rfPred\_scores\_motor    *Motor of rfPred\_scores*

---

### Description

Motor of rfPred\_scores

### Usage

```
rfPred_scores_motor(variant_list, data, index, all.col, file.export, n.cores)
```

### Arguments

variant_list	Variants list in a data.frame containing 4 or 5 columns: chromosome number, hg19 genomic position on the chromosome, reference nucleotid, variant nucleotid and uniprot protein identifier (optional)
data	Path to the compressed TabixFile, either on the server (default) or on the user's computer
index	Path to the index of the TabixFile, either on the server (default) or on the user's computer
all.col	TRUE to return all available information, FALSE to return a more compact result (the most informative columns, see Value)
file.export	Optional, name of the CSV file in which export the results (default is NULL)
n.cores	number of cores to use when scanning the TabixFile, can be efficient for large request (default is 1)

### Value

see the [rfPred\\_scores](#) function

### Note

This function is called by the rfPred\_scores S4 method

---

variant_list_Y	<i>Toy example of data.frame</i>
----------------	----------------------------------

---

**Description**

Toy example of data.frame

**Format**

A data frame with 5 observations on the following 5 variables:

chr Chromosome number (only Y in this example)

pos Position on the chromosome (numeric)

ref Referent nucleotid (A, C, G or T)

alt Alteration nucleotid (A, C, G or T)

uniprot Uniprot protein identifier (factor)

# Index

## \* data

example\_GRanges, [2](#)  
variant\_list\_Y, [7](#)

example\_GRanges, [2](#)

rfPred-package, [2](#)

rfPred\_scores, [3](#), [6](#)

rfPred\_scores, character-method  
(rfPred\_scores), [3](#)

rfPred\_scores, data.frame-method  
(rfPred\_scores), [3](#)

rfPred\_scores, GRanges-method  
(rfPred\_scores), [3](#)

rfPred\_scores\_motor, [6](#)

variant\_list\_Y, [7](#)