

# Package ‘proBAMr’

October 12, 2016

**Type** Package

**Title** Generating SAM file for PSMs in shotgun proteomics data

**Version** 1.6.0

**Date** 2016-03-31

**Author** Xiaojing Wang

**Maintainer** Xiaojing Wang <xiaojing.wang@vanderbilt.edu>

**Description** Mapping PSMs back to genome. The package builds SAM file from shotgun proteomics data. The package also provides function to prepare annotation from GTF file.

**License** Artistic-2.0

**Depends** R (>= 3.0.1), IRanges, AnnotationDbi

**Imports** GenomicRanges, Biostrings, GenomicFeatures, rtracklayer

**Suggests** RUnit, BiocGenerics

**biocViews** Proteomics, MassSpectrometry, Software, Visualization

**NeedsCompilation** no

## R topics documented:

PrepareAnnotationGENCODE . . . . .	1
PSMtab2SAM . . . . .	2

<b>Index</b>	<b>4</b>
--------------	----------

---

PrepareAnnotationGENCODE  
*prepare annotation from GENCODE*

---

## Description

prepare the annotation from GENCODE. Download GTF and FASTA files from GENCODE ftp first. Read introduction for more information.

**Usage**

```
PrepareAnnotationGENCODE(gtffile, CDSfasta, pepfasta, annotation_path,
  dbsnp = NULL, splice_matrix = FALSE, COSMIC = FALSE, ...)
```

**Arguments**

gtffile	specify GTF file location.
CDSfasta	path to the fasta file of coding sequence.
pepfasta	path to the fasta file of protein sequence.
annotation_path	specify a folder to store all the annotations.
dbsnp	specify a snp dataset to be used for the SNP annotation, default is NULL. (e.g. "snp135")
splice_matrix	whether generate a known exon splice matrix from the annotation. this is not necessary if you don't want to analyse junction results, default is FALSE.
COSMIC	whether to download COSMIC data, default is FALSE.
...	additional arguments

**Value**

several .RData files containing annotations needed for further analysis.

**Author(s)**

Xiaoqing Wang

**Examples**

```
gtffile <- system.file("extdata", "test.gtf", package="proBAMr")
CDSfasta <- system.file("extdata", "coding_seq.fasta", package="proBAMr")
pepfasta <- system.file("extdata", "pro_seq.fasta", package="proBAMr")
annotation_path <- tempdir()
PrepareAnnotationGENCODE(gtffile, CDSfasta, pepfasta,
  annotation_path, dbsnp=NULL,
  splice_matrix=FALSE, COSMIC=FALSE)
```

---

PSMtab2SAM

*Generate SAM files from PSMs.*

---

**Description**

Generate SAM files from confident peptide-spectrum-matches (PSMs).

**Usage**

```
PSMtab2SAM(passedPSM, XScolumn = "mvh", exon_anno, proteinseq, procodingseq,
  ...)
```

**Arguments**

passedPSM	a data frame of PSMs passed FDR.
XScolumn	specify the column which represents the matching score.
exon_anno	a dataframe of exon annotations.
proteinsseq	a dataframe containing protein ids and protein sequences.
procodingseq	a data frame containing coding sequence for each protein.
...	additional arguments

**Value**

a dataframe containing

**Author(s)**

Xiaojing Wang

**Examples**

```
load(system.file("extdata/GENCODE", "exon_anno.RData", package="proBAMr"))
load(system.file("extdata/GENCODE", "proseq.RData", package="proBAMr"))
load(system.file("extdata/GENCODE", "procodingseq.RData",
  package="proBAMr"))
options(stringsAsFactors=FALSE)
passedPSM <- read.table(system.file("extdata", "passedPSM.tab",
  package="proBAMr"), sep='\t', header=TRUE)
SAM <- PSMtab2SAM(passedPSM, XScolumn='mvh', exon, proteinsseq,
  procodingseq)
write.table(SAM, file=paste(tempdir(), '/test.sam', sep=''),
  sep='\t', quote=FALSE, row.names=FALSE, col.names=FALSE)
```

# Index

PrepareAnnotationGENCODE, [1](#)  
PSMtab2SAM, [2](#)