

Package: pgen2gds (via r-universe)

October 6, 2026

Type Package

Title Format Conversion between PLINK2 PGEN and GDS

Version 0.99.4

Date 2026-09-05

Depends methods, gdsfmt (>= 1.24.0)

Imports SeqArray (>= 1.49.6), pgenlibr

LinkingTo gdsfmt

Suggests parallel, digest, crayon, GenomicRanges, testthat (>= 3.0.0),
knitr, rmarkdown, BiocStyle, BiocGenerics

Description Provides functions to convert files from the PLINK2 pgen
format to SeqArray GDS, and from SeqArray GDS to the PLINK2
pgen format.

License GPL-3 + file LICENSE

VignetteBuilder knitr

BugReports <https://github.com/zhengxwen/pgen2gds/issues>

URL <https://github.com/zhengxwen/pgen2gds>

biocViews Infrastructure, DataImport, Genetics

Config/pak/sysreqs zlib1g-dev

Repository <https://bioc.r-universe.dev>

Date/Publication 2026-09-06 01:48:18 UTC

RemoteUrl <https://github.com/bioc/pgen2gds>

RemoteRef HEAD

RemoteSha 0988b4a99919bb20507fff6c410b1075ab1c5dd5

Contents

seqGDS2PGEN	2
seqPGEN2GDS	3
seqReadPVAR	5

Index	7
--------------	----------

seqGDS2PGEN

*Convert SeqArray GDS to PLINK2 PGEN files***Description**

Converts a SeqArray GDS file to PLINK2 pgen, pvar and psam files.

Usage

```
seqGDS2PGEN(gdsfile, out.fn,
  write.rsid=c("auto", "annot_id", "chr_pos_ref_alt"),
  half.call=c("missing", "haploid"), use.phase=TRUE, chr_prefix="",
  verbose=TRUE)
```

Arguments

<code>gdsfile</code>	a SeqVarGDSClass object or a file name of SeqArray GDS
<code>out.fn</code>	the file name prefix of the output PLINK2 files; <code>out.fn.pgen</code> , <code>out.fn.pvar</code> and <code>out.fn.psam</code> are created
<code>write.rsid</code>	"auto": use annotation/id and fall back to "chr_pos_ref_alt" when the ID is missing or "."; "annot_id": use annotation/id only; "chr_pos_ref_alt": use "chromosome_position_ref_alt"
<code>half.call</code>	how to handle a genotype with one missing allele (a half call or a haploid call): "missing" (default) codes it as missing, "haploid" codes it as homozygous for the observed allele
<code>use.phase</code>	if TRUE, write the phase information of heterozygous genotypes to the pgen file
<code>chr_prefix</code>	the prefix of chromosome codes, e.g., "chr"; it can be used to restore the prefix removed by <code>ignore.chr.prefix</code> in seqPGEN2GDS
<code>verbose</code>	if TRUE, show information

Details

The sample and variant selections set by [seqSetFilter](#) are respected, so `seqGDS2PGEN()` can export a subset of the GDS file. The ploidy has to be 2. Dosages are not exported.

The pgen file uses the PLINK2 storage mode 0x10. Each variant record is stored in the smallest of the dense 2-bit, 1-bit and difference-list encodings, following the thresholds used by PLINK2, and multiallelic genotypes and hardcall phase are stored in the corresponding auxiliary data tracks. The LD-based encodings of PLINK2 (differences from the previous variant) are not used, so the pgen file can be slightly larger than the file generated by PLINK2 itself, especially for datasets with many common variants in strong linkage disequilibrium. If the smallest file size is required, the files can be re-encoded by PLINK2, e.g., `plink2 --pfile out.fn --make-pgen --out new.fn`.

The pgen format keeps phase information for heterozygous genotypes only, so the phase flags of homozygous and haploid genotypes in the GDS file are ignored without loss of information.

The pvar file has the columns #CHROM, POS, ID, REF and ALT, where REF and ALT are taken from the GDS variable allele. If the chromosome codes are not recognized by PLINK2 (e.g., non-standard contig names), `--allow-extra-chr` may be needed when the files are loaded in PLINK2.

The psam file has the column IID from `sample.id`, and the columns FID, PAT, MAT, SEX and PHEN01 when the GDS variables `sample.annotation/family`, `sample.annotation/father`, `sample.annotation/mother`, `sample.annotation/sex` and `sample.annotation/phenotype` exist (the same variables as created by [seqBED2GDS](#) and [seqPGEN2GDS](#)).

Value

Return the file names of the pgen, pvar and psam files with absolute paths.

Author(s)

Xiuwen Zheng

References

<https://www.cog-genomics.org/plink/2.0/>

See Also

[seqPGEN2GDS](#), [seqGDS2BED](#), [seqGDS2VCF](#)

Examples

```
pgen_fn <- system.file("extdata", "plink2_gen.pgen", package="pgen2gds")

# PGEN to GDS
seqPGEN2GDS(pgen_fn, out.gdsfn="test.gds")

# GDS to PGEN
seqGDS2PGEN("test.gds", "test_out")

head(seqReadPVAR("test_out.pvar"))

# delete the temporary files
unlink(c("test.gds", "test_out.pgen", "test_out.pvar", "test_out.psam"),
      force=TRUE)
```

seqPGEN2GDS

Reformat PLINK2 PGEN files

Description

Reformats PLINK2 pgen files to GDS format.

Usage

```
seqPGEN2GDS(pgen.fn, pvar.fn=NULL, psam.fn=NULL, out.gdsfn,
  compress.geno="LZMA_RA", compress.annot="LZMA_RA", variant.sel=NULL,
  sample.sel=NULL, start=1L, count=NA_integer_,
  ignore.chr.prefix=c("chr", "0"), reference=NULL, optimize=TRUE,
  digest=TRUE, parallel=FALSE, balancing=TRUE, verbose=TRUE)
```

Arguments

<code>pgen.fn</code>	a file name for the pgen file
<code>pvar.fn</code>	a file name for the pvar file, or NULL to use the default
<code>psam.fn</code>	a file name for the psam file, or NULL to use the default
<code>out.gdsfn</code>	the file name of output GDS file
<code>compress.geno</code>	the compression method for "genotype"; optional values are defined in the function <code>add.gdsn</code>
<code>compress.annot</code>	the compression method for the GDS variables, except "genotype"; optional values are defined in the function <code>add.gdsn</code>
<code>variant.sel</code>	NULL for no variant selection, a logical vector or a numeric vector to specify the variant selection
<code>sample.sel</code>	NULL for no sample selection, a logical vector or a numeric vector to specify the sample selection
<code>start</code>	the starting variant if importing part of the pgen file
<code>count</code>	the maximum count of variant if importing part of the pgen file, <code>NA_integer_</code> or any non-positive value indicates importing to the end
<code>ignore.chr.prefix</code>	a vector of character, indicating the prefix of chromosome which should be ignored, e.g., "chr"; it is not case-sensitive
<code>reference</code>	genome reference, like "GRCh37", "GRCh38"; it is not specified if <code>reference=NULL</code>
<code>optimize</code>	if TRUE, optimize the access efficiency by calling cleanup.gds
<code>digest</code>	a logical value (TRUE/FALSE) or a character (e.g., "md5"); add hash codes to the GDS file if TRUE or a digest algorithm is specified
<code>parallel</code>	FALSE (serial processing), TRUE (parallel processing), a numeric value indicating the number of cores, or a cluster object for parallel processing; <code>parallel</code> is passed to the argument <code>cl</code> in seqParallel , see seqParallel for more details
<code>balancing</code>	whether to perform workload balancing or not, only applicable when multiple cores are used; if NA, use TRUE as a default until <code>getOption("seqarray.balancing")</code> is set and not TRUE
<code>verbose</code>	if TRUE, show information

Details

The sample IDs are taken from the column IID of the psam file. The psam columns FID, PAT, MAT and SEX are stored in the GDS variables `sample.annotation/family`, `sample.annotation/father`, `sample.annotation/mother` and `sample.annotation/sex` (coded as "M", "F" or "") if they exist, and the first phenotype column is stored in `sample.annotation/phenotype` (the same variables as created by [seqBED2GDS](#)). Dosages and phase information in the pgen file are not imported.

Value

Return the file name of SeqArray GDS file with an absolute path.

Author(s)

Xiuwen Zheng

References

<https://www.cog-genomics.org/plink/2.0/>

See Also

[seqGDS2PGEN](#), [seqReadPVAR](#)

Examples

```
pgen_fn <- system.file("extdata", "plink2_gen.pgen", package="pgen2gds")  
  
seqPGEN2GDS(pgen_fn, out.gdsfn="test.gds")  
  
# delete the temporary file  
unlink("test.gds", force=TRUE)
```

seqReadPVAR	<i>Read PLINK2 pvar file</i>
-------------	------------------------------

Description

Read PLINK2 pvar file for variants

Usage

```
seqReadPVAR(pvar, sel=NULL)
```

Arguments

pvar	a file name of a pvar file (from NewPvar), or a pvar object, which can be queried for variant IDs and allele codes
sel	NULL, a logical vector or a numeric vector for specifying the variants; NULL for including all variants

Value

Return a data frame with the columns chrom, pos, allele and rsid.

Author(s)

Xiuwen Zheng

References

<https://www.cog-genomics.org/plink/2.0/>

See Also

[seqPGEN2GDS](#), [seqGDS2PGEN](#)

Examples

```
pvar_fn <- system.file("extdata", "plink2_gen.pvar", package="pgen2gds")  
head(seqReadPVAR(pvar_fn))
```

Index

* **GDS**

- seqGDS2PGEN, [2](#)
- seqPGEN2GDS, [3](#)
- seqReadPVAR, [5](#)

* **PLINK2**

- seqGDS2PGEN, [2](#)
- seqPGEN2GDS, [3](#)
- seqReadPVAR, [5](#)

* **utilities**

- seqGDS2PGEN, [2](#)
- seqPGEN2GDS, [3](#)
- seqReadPVAR, [5](#)

cleanup.gds, [4](#)

NewPvar, [5](#)

seqBED2GDS, [3](#), [4](#)

seqGDS2BED, [3](#)

seqGDS2PGEN, [2](#), [5](#), [6](#)

seqGDS2VCF, [3](#)

seqParallel, [4](#)

seqPGEN2GDS, [2](#), [3](#), [3](#), [6](#)

seqReadPVAR, [5](#), [5](#)

seqSetFilter, [2](#)

SeqVarGDSClazz, [2](#)