Package 'lodGWAS'

November 30, 2015

Type Package

Title Genome-Wide Association Analysis of a Biomarker Accounting for Limit of Detection

Version 1.0-7

Date 2015-11-10

Author Ahmad Vaez, Ilja M. Nolte, Peter J. van der Most

Maintainer Ilja M. Nolte <i.m.nolte@umcg.nl>

Depends R (>= 3.0.0)

Imports survival, rms, stats, utils

Description Genome-wide association (GWAS) analyses of a biomarker that account for the limit of detection.

License GPL (>= 3)

NeedsCompilation no

Repository CRAN

Date/Publication 2015-11-30 13:35:40

R topics documented:

lodGWAS-package 1
lod_GWAS 2
lod_QC 7

Index **10**

`lodGWAS-package` *Genome-Wide Association Analysis of a Biomarker Accounting for Limit of Detection*

Description

Statistical analysis of a biomarker is often complicated because the detection range of the assay of the biomarker is restricted. The limits of detection (LOD) are the floor and/or ceiling values of the biomarker that can be accurately measured by a particular assay type. Any value of the biomarker beyond the range of LOD, either smaller than the lower LOD or larger than the upper LOD, cannot be determined accurately. Those observations, so-called non-detects (NDs), cannot simply be excluded from the analysis, because NDs are not 'missing at random'. They can be considered as censored data, and can therefore be best analyzed by using statistical methods for survival analysis.

lodGWAS is a flexible package for running genome-wide association analysis of a biomarker that accounts for the problem of limit of detection of the assay. It treats non-detected values as censored, and performs a parametric survival analysis on the phenotype of interest.

The analysis itself is carried out by the function `lod_GWAS`. A second function, `lod_QC`, can be used to check if the coding of LOD values in the phenotype file is correct.

Details

Package:	lodGWAS
Type:	Package
Version:	1.0-7
Date:	2015-11-10
License:	GPL (>= 3)

Author(s)

Ahmad Vaez, Ilja M. Nolte, Peter J. van der Most

Maintainer: Ilja M. Nolte <i.m.nolte@umcg.nl>

`lod_GWAS` *Genome Wide Association Analysis accounting for Limit of Detection*

Description

`lod_GWAS` enables the user to perform a Genome Wide Association Analysis (GWAS) of a biomarker accommodating the problem of Limit of Detection (LOD). This function performs a parametric survival analysis on the phenotype of interest that includes both measured and censored data.

`lod_QC` is automatically called within `lod_GWAS`, and its quality report will be saved in a separate text file.

Usage

```
lod_GWAS(phenofile, pheno_name,
        basic_model = NULL,
        dist = "gaussian",
        mapfile, genofile,
        outputfile,
        filedirectory = getwd(),
        outputheader = "QCGWAS", gzip_output = TRUE,
        lower_limit = NA, upper_limit = NA)
```

Arguments

phenofile	Either a data frame containing the phenotype (and covariate) values, or the filename (including the extension) of a data file containing the same. See below for information on the required format.
pheno_name	The name of the column in phenofile that contains the phenotype values.
basic_model	A formula (coded as a character string) describing the basic model, not including the genetic component. The covariates to be included into the analysis are mentioned within quotation marks separated by plus signs: for example, basic_model="sex+age". Please note that covariate names should exactly match the appropriate column names of phenotype file. The default is NULL, in which case the association is modelled without covariates.
dist	Assumed distribution for (raw or transformed) phenotype. The options are weibull, exponential, gaussian, logistic, lognormal and loglogistic. Default is gaussian. For more information, see the function psm of package rms.
mapfile	The file name of the genotype map file (including the file extension). See below for information on the required format.
genofile	The file name of the genotype dosage file (including the file extension). See below for information on the required format.
outputfile	The name for the output file.
filedirectory	The directory that contains the phenotype and genotype files and where the output file will be saved. Please note that R uses *forward* slash (/) where Windows uses backslash (\). The default setting is current R working directory.
outputheader	The output format of the analysis results file, to make it compatible with different software packages. The options are "QCGWAS", "GWAMA", "PLINK", "META", and "GenABEL". Default is "QCGWAS".
gzip_output	Logical; determines whether the output file is compressed. Default is TRUE.
lower_limit, upper_limit	Arguments passed to lod_QC. Specifying the limit of detection allows lod_QC to check if the phenotypes and outsideLOD columns have been coded correctly. Please note that these arguments are only used for a quality check. Any errors will be reported but *not* corrected. Default is NA.

Details

lod_GWAS is the main function of the package, and is capable of performing a genome-wide association study (GWAS) accommodating the problem of LOD. It treats non-detects as censored data, either left- or right-censored or both, and performs a parametric survival analysis on the phenotype of interest that includes both measured and censored values.

Value

`lod_GWAS` returns an invisible `NULL`. The real output are the association results (saved as `[output_file].txt`) and the log file generated by `lod_QC` (saved as `[output_file].txt.log`).

Input File Format

An analysis with `lod_GWAS` requires two files for the genotypes and one phenotype file. The files can be either space or tab delimited. The package also accepts files compressed in the gzip format (extension .gz).

Genotype Files

`lodGWAS` uses the PLINK dosage format for the genotype data. This means that two files are needed: one with the genotypes themselves (genotype dosage file), and one with the locations of the genetic variants (map file).

Genotype Dosage File

The genotype dosage file should contain a header line. The header line (first line) should be:

```
SNP A1 A2 FID1 IID1 FID2 IID2 ... FIDn IIDn
```

The first three columns must appear before the dosage data. The following columns are the family identifier (FID) and the individual identifier (IID) of individuals 1 to n. Thus, the number of columns of the header line should be exactly 3 + (2 x n_individuals).

The next lines contain the actual genetic data per individual, with each row corresponding to a genetic variant. The PLINK dosage format can be any of three formats: dosage, two-probabilities, or three-probabilities (see below). lodGWAS accepts all three formats and will automatically recognize whether there are one (dosage), two (two-probabilities), or three (three-probabilities) columns per individual. In case of any other format it will report that it cannot recognize the format and will not run.

Dosage format

A dosage is provided in one column per individual. Each dosage is a number between 0 and 2. A dosage of 0, 1, or 2 means that the individual is homozygous for the A2 allele, heterozygous, or homozygous for the A1 allele, respectively. When the genetic dataset is expanded using imputation, non-integer values are also possible, and are defined as the weighted sum of genotype probabilities (i.e. 0 x prob(A2/A2) + 1 x prob(A1/A2) + 2 x prob(A1/A1)). The number of columns of the (non-header) lines in a genotype file in dosage format should be exactly 3 + n_individuals.

Example of the dosage format:

```
      SNP  A1  A2   FID1 IID1 FID2 IID2 FID3 IID3
   rs0001  A   C    0.08      0.72      1.99
```

Two-probabilities format

Two numbers, representing the probabilities of the A1/A1 and A1/A2 genotypes, respectively. The probability of A2/A2 equals 1 minus the sum of Prob(A1/A1) and Prob(A1/A2). Each probability is a number between 0 and 1. The number of columns of the (non-header) lines in a genotype file in two-probabilities format should be exactly 3 + (2 x n_individuals).

Example of the two-probabilities format:

```
     SNP  A1  A2   FID1 IID1   FID2 IID2
  rs0001   A   C   0.97 0.02   0.88 0.10
```

Three-probabilities format

Three numbers, representing the probabilities of the A1/A1, A1/A2, and A2/A2 genotypes, respectively. Each probability is a number between 0 and 1, and the three probabilities per genetic variant per individual should add up to 1. The number of columns of the (non-header) lines in a genotype file in three-probabilities format should be exactly 3 + (3 x n_individuals).

Example of the three-probabilities format:

```
     SNP  A1  A2   FID1 IID1       FID2 IID2
  rs0001   A   C   0.97 0.02 0.01  0.88 0.10 0.02
```

Genotype Map File

The genotype map file contains the locations of the genetic variants, with each row of the file corresponding to a variant. It must contain four columns:

- Chromosome (1-22, X, Y or 0 if unspecified)
- Marker ID (identifier of the genetic variant)
- Genetic distance (Morgan, this is not used by `lod_GWAS`, so the actual value doesn't matter)
- Physical position (base-pair position)

Note: unlike the other input files, the map file has *no* header line.

Phenotype File

The phenotype file is a text file containing the non-genetic data, with each row of the file corresponding to an individual. It must meet the following requirements:

- The file must have a header line.
- it must contain the following variables: family ID, individual ID, phenotype, and outside-LOD (which indicates whether the phenotype is measered, or left- or right-censored). Other columns, e.g. for covariates, are optional.
- The header (name) of columns for family ID, individual ID, and outsideLOD must be `FID`, `IID`, and `outsideLOD`, respectively (note that R is case sensitive). The other columns (phenotype and cov1 to covN) can have any arbitrary name.

The order of the rows (samples) or columns is not important.

Column descriptions of phenotype file

- FID: family identifier of the individual. It must match the FIDs in the genotype dosage file.
- IID: the unique identifier of the individual within each family. It must match the IIDs in the genotype dosage file.

- Phenotype: the phenotype or trait of interest, which can be any numeric value. See below for a few considerations regarding the phenotype.
- outsideLOD: The variable `outsideLOD` indicates whether the phenotype value is within or beyond the range of LOD. It must be to be coded as `0` if phenotype > upper LOD; `1` if phenotype is within the detection interval; and `2` if phenotype < lower LOD. Values other than `0`, `1`, or `2` are not accepted.
- cov1 to covN: covariate 1 to covariate N. The phenotype file can contain as many covariates as necessary. Some examples are: age, sex, BMI, smoking status, medication, population stratification parameters (principal components), dosage data of a particular genetic variant (for conditional analysis), study center, etc.

A few considerations regarding the phenotype

Please pay particular attention to these instructions, as failing to heed them may cause invalid results.

1) The user must carefully distinguish between two types of missing phenotype: missing and censored values. Any mix-up between these two types will yield incorrect results.

2) *Missing phenotype values* are those phenotypes that are missing for any reason other than being beyond the LOD. They are considered as real missing (at random). These values must be coded as `NA` in both `Phenotype` and `outsideLOD` columns.

3) *Censored phenotype values* are NDs, i.e. measurements that fall beyond the LOD of the assay. NDs are not real missing values, since they do provide information about the distribution of the phenotype. Any ND that is below the lower LOD should be changed to the value of the lower LOD (and the corresponding outsideLOD value should set to `2`). Any ND that is above the upper LOD should be changed to the value of the upper LOD (and the corresponding outsideLOD value should be set to `0`). NDs should *NOT* be coded as missing (`NA`). `lodGWAS` can handle multiple lower and upper LOD levels (e.g. as a result from different assays used to measure the biomarker) in a single file. In that case the phenotype of an ND should be changed to the lower/upper LOD level of the assay type used for that individual.

4) The column phenotype can be either raw or transformed values of the phenotype. Please take care that NDs (whose phenotype value equals the LOD) must also be transformed appropriately.

Output File Format

Column descriptions of the output file (as per default settings, with `outputheader="QCGWAS"`) are as following:

- MARKER: marker ID (identifier of the genetic variant) as specified in the genotype input files
- CHR: chromosome as specified in the genotype map file
- POSITION: physical position (base-pair position) as specified in the genotype map file
- OTHER_ALL: non-effect allele (non-coded allele)
- EFFECT_ALL: effect allele (coded allele)
- N_TOTAL: total sample size, including all NDs as well as valid measured values
- N_VALID: the sample size of valid measured values (excluding all NDs). This is useful if the user wants to know the percentage of NDs to the total sample size.
- EFF_ALL_FREQ: effect allele frequency

- EFFECT: effect size (beta) of effect allele
- STDERR: standard error of effect allele
- PVALUE: p-value of association
- IMP_QUALITY: imputation quality of the genetic variant

If another output format is chosen, the same columns will be present in the output file, but with header names as required by the specified software program.

Note

GWAS analysis will not be performed: 1) on rare genetic variants (with allele frequency <0.001 or >0.999), and 2) on badly imputed genetic variants (with imputation quality score < 0.01). Those genetic variants will be included in the output file, but the association results will be NA.

Examples

```
# For use in this example, the 3 Sample files in the
# extdata folder of the lodGWAS library will be copied
# to your current R working directory

## Not run:

file.copy(from = file.path(system.file("extdata", package = "lodGWAS"), "Sample_geno.dose"),
               to = getwd(), overwrite = FALSE, recursive = FALSE)
file.copy(from = file.path(system.file("extdata", package = "lodGWAS"), "Sample_geno.map"),
               to = getwd(), overwrite = FALSE, recursive = FALSE)
file.copy(from = file.path(system.file("extdata", package = "lodGWAS"), "Sample_pheno.txt"),
               to = getwd(), overwrite = FALSE, recursive = FALSE)

lod_GWAS(phenofile = "Sample_pheno.txt", pheno_name = "outcome1",
         basic_model = "sex",
         mapfile = "Sample_geno.map", genofile = "Sample_geno.dose",
         outputfile = "Sample_output.txt", gzip_output = FALSE,
         lower_limit = 0.1, upper_limit = 2)

## End(Not run)
```

---

lod_QC *Checking the coding of phenotype and outsideLOD values*

Description

It is important that the phenotype and outsideLOD values in the phenotype file are coded correctly. The function lod_QC checks the quality of the phenotype file, and provides a number of descriptive statistics about the phenotype, as well as warnings about suspected problems.

Usage

```
lod_QC(phenofile,
      pheno_name,
      filedirectory = getwd(),
      outputfile = "lodQC",
      lower_limit = NA, upper_limit = NA,
      stop_if_error = FALSE,
      reprint_warnings = FALSE)
```

Arguments

phenofile	either a data frame containing the phenotype (and covariate) values, or the filename (including the extension) of a data file containing the same. For details on the required format of the phenotype file, see the section on File Format in the description of lod_GWAS.
pheno_name	the name of the column in phenofile contain the phenotype values.
filedirectory	the directory that contains the phenotype file and where the output file will be saved. Please note that R uses *forward* slash (/) where Windows uses backslash (\). The default setting is current R working directory.
outputfile	the name for the output file.
lower_limit, upper_limit	two numeric values specifying the lower and upper LOD of the assay. Defaults are NA, in which case lod_QC only checks for gaps between measured and censored values.
stop_if_error	logical; determines whether the function aborts or continues when it encounters errors in the dataset. Default is FALSE.
reprint_warnings	logical; this argument is used by lod_GWAS and isn't relevant for users. Setting it to TRUE will cause certain errors to be reported via warning (in addition to the console and the log file). Default is FALSE.

Details

The user can specify values for the lower and upper LOD. The function will then compare the values in the column outsideLOD with the phenotype values, conditional on the user-specified LOD. For details on the required format of the phenotype file, see the section on File Format in the description of lod_GWAS.

Also, lod_QC will check if there is a gap between the smallest/largest values within the LOD (i.e. truly measured values, where outsideLOD is 1), and the censored values. If the censored values are far smaller/larger than smallest/largest measured value, lod_QC will print a warning. An example of such an erroneous coding of censored values is when the phenotype values below the lower LOD have been set to -9, while the measured values are >0.

Value

The function lod_QC returns an invisible NULL. The real output is a log file containing information on the distribution of the phenotype, and any problems it encountered. These warnings are also displayed in the R console.

Note

The function lod_QC only checks and reports. It does *not* correct the phenotype or outsideLOD values.

Also, the function assumes that there is a single lower and/or upper LOD. If multiple LODs are used in the file, the function may produce warnings when both censored and real values do not match the specified limits. In that case it is up to the user to ensure the quality of the phenotype file.

See Also

lod_GWAS for the complete GWAS function.

Examples

```
phenos <- data.frame(FID = c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10),
                     IID = c("female1", "male2", "male3", "female4", "female5",
                             "male6", "female7", "male8", "male9", "male10"),
                     sex = c(0, 1, 1, 0, 0, 1, 0, 1, 1, 1),
                     outcome1 = c(0.3, 0.5, 0.9, 0.7, 2, 2, 0.1, 1.1, 2, 0.7),
                     outsideLOD = as.integer(c(1, 1, 1, 1, 1, 0, 2, 1, 0, 1)),
                     stringsAsFactors = FALSE)

lod_QC(phenofile = phenos, pheno_name = "outcome1",
       outputfile = "Sample_QC", lower_limit = 0.1, upper_limit = 2)
```

Index

∗Topic **package**
lodGWAS-package, 1

lod_GWAS, *2*, 2, *8, 9*
lod_QC, *2–4*, 7
lodGWAS *(*lodGWAS-package*)*, 1
lodGWAS-package, 1

warning, *8*