EasyQC

DESCRIPTION

This software-package provides functions to perform QC checks for genome-wide association studies.

Author: Thomas Winkler thomas.winkler@klinik.uni-regensburg.de
Version: 9.0
License: GPL v3
Citation: If you are using EasyX, please cite "Winkler et al.: *Quality control and conduct of genome-wide association meta-analyses.* Nature Protocols 2014" and (if possible) reference our website www.genepi-regensburg.de/easyqc

Date: 2014-09-18

INSTALLATION

Please install the R-package EasyQC using

```
> install.packages("/path2tarball/EasyQC_9.0.tar.gz")
```

Dependencies: R-packages "Cairo", "plotrix", "data.table".

USAGE

The software can be started using the EasyQC function from the R command-line. The Function takes an EasyQC config/script (ECF-file) file and performs all steps defined in the ECF-file.

Example:

```
> library(EasyQC)
> EasyQC("/path2ecffile/test.ecf")
```

General Structure of an ecf-file:

```
#####################################################

<EasyQC configuration parameters (functions DEFINE and EASYIN)

START EASYQC

<EasyQC scripting interface (EasyQC functions)>

STOP EASYQC

#####################################################
```

Each ecf-file takes one set of configuration parameters and one set of scripting function, i.e. START EASYQC and STOP EASYQC may only be used once with an ecf-file. It is not allowed to start over with a different input and pipeline after closing the evaluation with STOP EASYQC.

EasyQC Functions

EASY CONFIGURATION

What files should be processed? Where to save the results?

DEFINE

Set parameters that will be valid for all input files except if they will be overwritten for any specific file in the EASYIN statement.

Input:

PARAMETER	*DESCRIPTION*
--strMissing	Missing value character. Optional. Default: NA
--strSeparator	Column separator. Optional. Default: WHITESPACE Please use: [WHITESPACE|COMMA|TAB|SPACE]
--acolIn	Array of Input columns. Optional. By default all columns will be read. Please use: Column names separated by ';'. This can be used to define the input columns for fast reading, renaming and exclusion of columns. If set, only the columns stated will be read. All columns stated here must be present in the input. The function is case-sensitive.
--acolInClasses	Array of Input column classes. Optional. By default all columns will be read using best guess classes estimated from the first 10 rows of the input. Please use: [character|numeric|double|integer|logical] separated by ';' respectively for columns defined at --acolIn.
--acolNewName	Array of new input columns. Optional. Please use: New column names separated by ';' respectively for columns defined at --acolIn. If set, the column names stated at --acolIn will be renamed to the names stated here.
--pathOut	Path to save all output. Optional. Default: Working directory (getwd()).

Example:

```
DEFINE      --strMissing .
            --strSeparator TAB
            --acolIn SNP;A1;A2;EAF;P;N
            --acolInClasses character;character;character;numeric;numeric;integer
            --acolNewName MarkerName;Allele1;Allele2;Freq;pvalue;samplesize
            --pathOut /path2outputfolder/out
```

EASYIN

Set the input files.

Inputs:

PARAMETER	*DESCRIPTION*
-- fileIn	Path to Input file.
-- fileInShortName	Short name of the input file that will be used in logs/reports/plots. Optional. Default: The filename of the input.
-- fileInTag	Set a tag for the input file, e.g. MEN and WOMEN for a men- and a women-specific file respectively. The tag will be added to all input columns, which may be helpful if the input data is supposed to be merged by MERGEEASYIN.
--astrSetNumCol	String of the format "<NEWCOL1>=<VAL1>;<NEWCOL2>=<VAL2>;…" that can be used to add numeric columns with constant values, e.g. N0=100;N1=2000 add columns N0 and N1 for which all SNPs will have the values 100 and 2000 respetively.

In addition to these parameters, the DEFINE parameters --strMissing, --strSeparator, --acolIn, --acolInClasses, --acolNewName can be set at EASYIN as well, which will overwrite the DEFINE statement specifically for the specified file.

Example:

```
EASYIN      --fileIn /path2input/file1.txt
            --fileInShortName FILE1
            --fileInTag MEN
            --astrSetNumCol N=1300
```

GENERAL FUNCTIONS

ADDCOL

Add columns to input.

Input:

PARAMETER	*DESCRIPTION*
--rcdAddCol	R-Code expression to calculate the added column. Result will be added by cbind() to the input.
--colOut	Name of the added column.
--blnOverwrite	Boolean value to specify whether existing column should be overwritten. Optional. Default: 1 (Existing column will be overwritten) Please use: [0|1].

Example:

```
ADDCOL        --rcdAddCol pmin(EAF*N,(1-EAF)*N,na.rm=TRUE)
              --colOut MAC
```

Output:

Column <colOut>, (e.g. 'MAC' in the example) will be added to the data-set.

ADJUSTALLELES

Adjust allele directions according to reference allele directions.

Input:

PARAMETER	*DESCRIPTION*
--colRefStrand	Column name of the reference strand. Optional. If this is not defined, all reference strand will be set to "+".
--colRefA1	Column name of the reference Allele1.
--colRefA2	Column name of the reference Allele2.
--colInStrand	Column name of the input strand. Optional. If this is not defined, all input strand will be set to "+".
--colInA1	Column name of the input Allele1.
--colInA2	Column name of the input Allele2.
--colInFreq	Column name of the input allele-frequency. In case the allele direction will be switched in order to match the reference alleles, this column will be adjusted for the respective SNPs: FreqAdjusted=1-Freq.
--acolInFreq	Array of multiple frequency columns that refer to the given alleles. In case the allele direction will be switched in order to match the reference alleles, ALL these columns will be adjusted for the respective SNPs by changing the direction: FreqAdjusted = 1-Freq.
--colInBeta	Column name of the input effect estimate. In case the allele direction will be switched in order to match the reference alleles, this column will be adjusted for the respective SNPs by changing the effect direction: BetaAdjusted= - colInBeta.
--acolInBeta	Array of multiple beta columns that refer to the given alleles. In case the allele direction will be switched in order to match the reference alleles, ALL these columns will be adjusted for the respective SNPs by changing the effect direction: BetaAdjusted = - Beta.
--blnMetalUseStrand	Boolean value. If set to 1 (TRUE), the alleles will be switched according to metal's option "USESTRAND ON" (Willer, et al., 2010). Optional. Default: 0. Please use: [0|1] Please note that blnMetalUseStrand=0 is not fully identical with the metal option USESTRAND OFF. Please see below a detailed description / comparison of metal's USESTRAND and EasyQC's blnMetalUseStrand option. Optional. Default: 1. Please use [0|1]
--blnWriteMismatch	Boolean value to define whether allele mismatches between the input and reference will be written to a separate file in the output path. Mis-match definition: SNPs that carry valid alleles (defined as 'A','C,'G','T','I','D') but cannot be matched between input and reference: For example if a SNP is coded A/T in the input, and A/G in the reference file. Optional. Default: 1 Please use [0|1].
--blnRemoveMismatch	Boolean value to define whether allele mismatches between the input and reference will be removed from the input. Optional. Default: 0. Please use [0|1].
--blnWriteInvalid	Boolean value to define whether SNPs with invalid input allele codes (other 'A','C,'G','T','I','D') should be written to a separate file in the output path. Optional. Default: 1. Please use [0|1].

--blnRemoveInvalid	Boolean value to define whether SNPs with invalid input allele codes will be removed from the input. Optional. Default: 0. Please use: [0|1].
--blnWriteRefInvalid	Boolean value to define whether SNPs with invalid reference allele codes (other 'A','C,'G','T','I','D') should be written to a separate file in the output path. Optional. Default: 0. Please use: [0|1].
--blnRemoveRefInvalid	Boolean value to define whether SNPs with invalid reference allele codes will be removed from the input. Optional. Default: 0. Please use: [0|1].
--strTag	Tag for the function step that will be added to related variables in the REPORT (e.g. number of mismatching SNPs) and to related output (e.g. files written by --blnWriteInvalid 1) to ensure unique and easily recognizable file names and REPORT variable names.
--fileRef, --strRefSuffix .ref, --strInSuffix, --colInMarker, --colRefMarker, --blnInAll, --blnRefAll, --blnWriteNotInRef, --blnWriteNotInIn	Inherited MERGE parameters. See MERGE for a more detailed description. Parameter values are given if they differ from the default MERGE values.
--strMissing , --strSeparator, --acolIn, --acolInClasses, --acolNewName	Inherited DEFINE parameters. Can be used for --fileRef. See DEFINE for a more detailed description.

Inherited parameters can be used with ADJUSTALLELES directly.
In particular, setting --fileRef at ADJUSTALLELES can be helpful if the input does not (yet) contain reference alleles and if reference data needs to be merged from external data (this may replace an extra MERGE step in the ecf-file).
If the input already contains reference alleles, there is no need to use --fileRef or any other inherited parameters.

Example:

```
ADJUSTALLELES --colRefStrand Strand.ref
              --colRefA1 A1.ref
              --colRefA2 A2.ref
              --colInStrand Strand
              --colInA1 EffectAllele
              --colInA2 OtherAllele
              --acolInFreq EAF.men;EAF.women
              --acolInBeta BETA.men;BETA.women
              --blnMetalUseStrand 0
              --blnWriteMismatch 0
              --blnRemoveMismatch 0
              --blnWriteInvalid 0
              --blnRemoveInvalid 0
              --fileRef /path2ref/allelefreqreference.txt
                 --acolIn SNP;Strand;A1;A2;Freq1
                 --acolInClasses character;character;character;character;numeric
              --colRefMarker SNP
              --colInMarker MarkerName
```

Output:
The defined input alleles, frequency and betas will be aligned to the defined reference alleles.

REPORT VARIABLES	*DESCRIPTIPON*
Checked	Number of SNPs that carry valid and non-missing input and reference allele or strand information and thus are being considered for adjustment.
StrandChange	Number of SNPs with opposite strand (e.g. + in input and - in reference).
AlleleMatch	Number of SNPs with matching alleles, same direction and same strand (e.g. +AC in input and +AC in reference).
AlleleChange	Number of SNPs with matching alleles, switched direction and same strand (e.g. +AC in input and +CA in reference).
n4AlleleMatch	Number of non-palindromic SNPs with matching alleles, same direction and switched strand (e.g. +AC in input and +TG (-AC) in reference).
n4AlleleChange	Number of non-palindromic SNPs with matching alleles, switched direction and switched strand (e.g. +AC in input and +GT (-CA) in reference).
AlleleMismatch	Number of SNPs with allele missmatch (e.g. +AG in input and +AC in reference).
AlleleInMissing	Number of SNPs with missing input allele.
AlleleInInvalid	Number of SNPs with invalid input allele (other than A,C,G,T,I,D).
StrandInInvalid	Number of SNPs with invalid input strand (other than +,-).
AlleleRefMissing	Number of SNPs with missing reference allele.
AlleleRefInvalid	Number of SNPs with invalid reference allele (other than A,C,G,T,I,D).
StrandRefInvalid	Number of SNPs with invalid reference strand (other than +,-).
NotInRef, NotInIn	Inherited MERGE variables. Only present if --fileRef is used. Please see MERGE for a more detailed description.

FILE OUTPUTS	*DESCRIPTION*
*.mismatch.txt	If --blnWriteMisMatch 1, a separate file will be written to pathOut that contains all mismatching SNPs.
*. invalid.txt	If --blnWriteInvalid 1, a separate file will be written to pathOut that contains all SNPs with invalid input or reference alleles or strand.
*.notinref.txt, *.notinin.txt	Inherited MERGE output. Only present if --fileRef is used. Please see MERGE for a more detailed description.

If --strTag is defined, the <strTag> string value will be pasted to the report variable names and to the output filenames.

Details of allele adjustment – Metal vs EasyQC:

Meta-analysis software metal, option USESTRAND (Willer, et al., 2010):

USESTRAND OFF	*Disregards strand column.*	
	Study1=AC, Study2=TG:	maps A-T
	Study1=AT, Study2=AT:	maps A, disregards strand (if given)

USESTRAND ON *Uses strand column for palindromic SNPs.*
Study1=AC, Study2=TG: maps A-T (same as before)
Study1=+AT, Study2=-AT: changes strand of Study2 to +TA -> maps A-A

EasyQC option blnMetalUseStrand:

For both options (0/1), the EasyQC algorithm first sets missing or non-defined Strand to + and changes all '-' strand to '+' by switching alleles accordingly (A->T; T->A; C->G; G->C).

--blnMetalUseStrand 0 *Does NOT match non-palindromic SNPs on wrong strand (+AC vs +TG will become mismatch)*
Study1=+AC, Study2=+TG: mis-match
Study1=+AT, Study2=+TA: maps A
Study1=+AT, Study2=-TA: Changes strand of Study2 to +AT-> maps A-A

--blnMetalUseStrand 1 *Matches non-palindromic SNPs on wrong strand (+AC and +TG; maps A-T)*
Study1=+AC, Study2=+TG: maps A-T
Study1=+AT, Study2=-AT: Changes strand of study 2 to +TA -> maps A-A

Summary:

EasyQC's '--blnMetalUseStrand 1' and metal's 'USESTRAND ON' will produce identical results.
EasyQC's '--blnMetalUseStrand 0' and metal's 'USESTRAND OFF' differ

- for non-palindromic SNPs with non-matching Strand: Study1=+AC, Study2=+TG:
  EasyQC: labels SNP as mismatch (if blnMetalUseStrand 0; matched otherwise)
  metal: matches A-T
- for palindromic SNPs with non-matching Strand: Study1=+AT, Study2=-TA:
  EasyQC: Recodes strand of Study2 to +AT and matches A-A
  metal: matches A-A while disregarding Strand (which would be wrong if the strand coding is valid)

CALCULATE

Calculate values from input. Result will be written to the REPORT variable defined in --strCalcName and can be used by RPLOT subsequently.

Input:

PARAMETER	*DESCRIPTION*
--rcdCalc	R-Code expression to calculate the value. Needs to return a single value.
--strCalcName	Name of the REPORT variable to save the calculated value.

Example:

```
CALCULATE     --rcdCalc 2/median(SE,na.rm=T)
              --strCalcName num2overMedianSE
```

Output:

The input data-set remains unchanged.

REPORT VARIABLES	*DESCRIPTIPON*
<strCalcName>	The calculated value.

CLEAN

Exclude SNPs from input. Number of exclusion will be written to the REPORT variable defined in --strCleanName.

Input:

PARAMETER	*DESCRIPTION*
-- rcdClean	R-Code expression to exclude SNPs. Needs to return a Boolean array. TRUE values will be removed from the input.
-- strCleanName	Name of the REPORT variable to save the number of exclusions.
--blnWriteCleaned	Boolean value to define whether SNPs that are removed from the input should be written to a separate file in the output folder. Optional. Default: 1 Please use: [0|1].

Example:

```
CLEAN  --rcdClean (P<0 | P>1)
       --strCleanName numDropSNP_invalid_P
       --blnWriteCleaned 0
```

Output:

All SNPs that meet the defined cleaning criterion will be removed from the input and no more be present in the output data set.

REPORT VARIABLES	*DESCRIPTIPON*
<strCleanName>	Number of removed SNPs.

FILE OUTPUTS	*DESCRIPTION*
*.<strCleanName>.txt	If --blnWriteCleaned 1, a separate file that contains all removed SNPs.

CRITERION

Apply criterion to gwadata. SNPs that match the criterion will be written in a unique file in the output folder. The GWA data set remains unchanged. This is only to extract SNPs that match a specific criterion.
In addition the number of matches will be written into the report.

Input:

PARAMETER	*DESCRIPTION*
--rcdCrit	R-Code expression to define the criterion SNPs. Needs to return an array of Boolean values. TRUE value rows will be written to separate file in pathOut and number of matches will be written to the report.
--strCritName	Name of the REPORT variable to save the number of SNPs that fulfill the criterion.

Example:

```
CRITERION     --rcdCrit P<=5e-8
              --strCritName numSNP_gws
```

Output:

The input data-set remains unchanged.

REPORT VARIABLES	*DESCRIPTIPON*
<strCritName>	Number of SNPs that match the criterion.

FILE OUTPUTS	*DESCRIPTION*
*.<strCritName>.txt	This file is a subset of the gwadata and contains all SNPs that match the defined criterion.

EDITCOL

Edit columns of the input.

Input:

PARAMETER	*DESCRIPTION*
--rcdEditCol	R-Code expression to calculate the new values. Result will be written to column colEdit. It might be useful to use the ifelse statement.
--colEdit	Name of the edited column.

Example:

```
EDITCOL       --rcdEditCol ifelse(BETA==-9,NA,BETA)
              --colEdit BETA
```

In the example, all -9 values in column BETA will be replaced with NA.

Output:

Column <colEdit> will be updated with edited values.

EVALSTAT

Calculate descriptive statistics. Number of values, number of missing values, minimum, maximum, median, 25th percentile, 75 percentile, mean and standard deviation will be written to the REPORT.

Input:

PARAMETER	*DESCRIPTION*
--colStat	Evaluated column. Requested.
--strTag	Optional. Tag for the function step that will be added to related variables in the REPORT. Default: ''

Example:

```
EVALSTAT      --colStat P
              --strTag preQC
```

Output:

The input data-set remains unchanged.

REPORT VARIABLES	*DESCRIPTIPON*
<strTag>.<colStat>_num	Number of SNPs tested.
<strTag>.<colStat>_NA	Number of SNPs with missing values.
<strTag>.<colStat>_min	Minimum value.
<strTag>.<colStat>_max	Maximum value.
<strTag>.<colStat>_median	Median.
<strTag>.<colStat>_p25	25^{th} percentile.
<strTag>.<colStat>_p75	75^{th} percntile.
<strTag>.<colStat>_mean	Mean value.
<strTag>.<colStat>_sd	Standard devialtion.

EXTRACTSNPS

Extract set of SNPs from the input data-set.

Input:

PARAMETER	*DESCRIPTION*
--colInMarker	SNP column name of the input. Will be used for extraction.
--fileRef	Path to the reference data including SNPs that will be extracted.
--colRefMarker	SNP column name of the reference. Will be used for extaction.
--strTag	Tag for the function step that will be added to related variables in the REPORT (e.g. number of SNPs Not in Ref) and to related output (e.g. files written by --blnWriteNotInRef 1) to ensure unique and easily recognizable file names and REPORT variable names.

Example:

```
EXTRACTSNPS   --colInMarker MarkerName
              --fileRef /test/snps2extract.txt
              --colRefMarker rsID
              --strTag KnownSnps
```

Output:

The input data set remains unchanged.

REPORT VARIABLES	*DESCRIPTIPON*
<strTag>.numExtractMissing	Number SNP from the reference that are not present in the input.

FILE OUTPUTS	*DESCRIPTION*
*.<strTag>.extracted.txt	File containing the cut-out SNPs.

FILTER

Filter SNPs from input. Number of inclusion will be written to the REPORT variable defined in --strFilterName.

Input:

PARAMETER	*DESCRIPTION*
--rcdFilter	R-Code expression to include SNPs. Needs to return an array of Boolean values. TRUE value rows will be included (pass the filter).
--strFilterName	Name of the REPORT variable to save the number of inclusions.

Example:

```
FILTER --rcdFilter MAC>=3
       --strFilterName numSNP_MACget3
```

Output:

The output data set will only contain SNPs that pass the defined filter criterion.

REPORT VARIABLES	*DESCRIPTIPON*
<strFilterName>	Number of SNPs that pass the filter.

GC

Genomic control correction. The GC Lambda can be calculated from all SNPs or from a subset of SNPs (see –fileGcSnps); will be written to the report; and can optionally be applied to the data set (see --blnSuppressCorrection).

Input:

PARAMETER	*DESCRIPTION*
--colPval	Define P-Value column that will be used for calculating the GC lambda and for the correction.
--colSE	If defined, this Standard error column will be corrected as well.
--numLambda	If defined, this lambda will be used for performing the correction. Optional. Default: NA (causes EasyQC to calculate the lambda from the specified colPval)
--fileGcSnps	If defined, only SNPs from this file will be used for calculating the lambda. Optional. Default: NA (causes EasyQC to calculate the lambda from the full list of input SNPs)
--colGcSnpsMarker	If fileGcSnps is defined, please define here the Marker column name of the file. Optional. Default: NA (only required if fileGcSnps is specified)
--blnSuppressCorrection	Boolean value. If set to 1 (TRUE), the GC lambda will be calculated, but the actual correction will not be performed. If set to 0 (FALSE), the GC lambda will be calculated and the colPval and colSE (if specified) will be corrected. New columns with the suffix ".GC" will be added to the GWA data set. Optional. Default: 0; Please use: [0|1]
--colInMarker	If fileGcSnps is defined, please define here the Marker column name of the input. Required (if fileGcSnps is set).
--strTag	Optional. Tag for the function step that will be added to related variables in the REPORT.

Example:

```
GC      --colPval P
        --colSE SE
        --fileGcSnps /home/gcfile.txt
        --colGcSnpsMarker MarkerNameOfFileGcSnps
        --blnSuppressCorrection 0
        --colInMarker MarkerName
        --strTag preQC
```

In this example the lambda will be calculated on column P for all SNPs specified in fileGcSnps. Afterwards column P and SE will be GC-corrected using the estimated lambda and 2 new columns P.GC and SE.GC will be added to the input data set.

Output:

If **--**blnSuppressCorrection 0, the output data set will carry additional columns <colPval>.GC and (if --colSE defined) <colSE>.GC.

REPORT VARIABLES	*DESCRIPTIPON*

Lambda.<colPval>.GC	GC lambda.

GETCOLS

Extract columns. This removes all columns from the input that are not stated at --acolOut.

Input:

PARAMETER	*DESCRIPTION*
-- acolOut	Array of extracted columns. The data set will be reduced to the here stated columns. Please use: Column names separated by ';'.

Example:

```
GETCOLS        --acolOut MarkerName;P
```

Output:

The output data set will only contain columns defined at <acolOut>. All other columns will be removed.

GETNUM

Get number of SNPs that match a certain criterion. Number of matches will be written to the REPORT variable defined in --strGetNumName.
Please note that the data set itself remains unchanged. This is only to get the number of matches and to write them into the report.

Inputs:

PARAMETER	*DESCRIPTION*
--rcdGetNum	R-Code expression to derive the number of SNPs. Needs to return an array of Boolean values. All TRUE values will be counted.
--strGetNumName	Name of the REPORT variable to save the number of matches.

Example:

```
GETNUM --rcdGetNum abs(BETA)>5
       --strGetNumName numSNP_BETAgt5
```

Output:

The input data-set remains unchanged.

REPORT VARIABLES	*DESCRIPTIPON*
<strGetNumName>	The number of SNPs that meet the defined criterion rcdGetNum.

MERGE

Merge reference data (e.g. annotation data like chromosome and position) to each of the input data-sets.

Input:

PARAMETER	*DESCRIPTION*
--colInMarker	SNP column name of the input. Will be used for merging.
--fileRef	Path to the reference data set that will be added.
--colRefMarker	SNP column name of the reference. Will be used for merging.
--strInSuffix	Suffix that will be added to all input columns except to colInMarker. Optional. By default '.x' will be used for overlapping columns.
--strRefSuffix	Suffix that will be added to all reference columns except to colRefMarker. Optional. By default '.y' will be used for overlapping columns.
--blnInAll	Boolean value to define left inner/outer join. If set to 1 (TRUE), all SNPs from the input will be present in the merged data-set. If set to 0 (FALSE), only SNPs from the input will be present in the merged data-set that are also present in the reference.Optional. Default: 1. Please use: [0|1].
-- blnRefAll	Boolean value to define right inner/outer join. If set to 1 (TRUE), all SNPs from the reference will be present in the merged data-set. If set to 0 (FALSE), only SNPs from the reference will be present in the merged data-set that are also present in the input.Optional. Default: 0. Please use: [0|1].
--strTag	Tag for the function step that will be added to related variables in the REPORT (e.g. number of SNPs Not in Ref) and to related output (e.g. files written by --blnWriteNotInRef 1) to ensure unique and easily recognizable file names and REPORT variable names.
--blnWriteNotInRef	Boolean value to define whether SNPs from the input that are missed in the reference will be written to a separate file in the output path. Optional. Default: 0. Please use: [0|1].
-- blnWriteNotInIn	Boolean value to define whether SNPs from the reference that are missed in the input will be written to a separate file in the output path. Optional. Default: 0. Please use: [0|1].
--strMissing , --strSeparator, --acolIn, --acolInClasses, --acolNewName	Inherited DEFINE parameters. Can be used for --fileRef. See DEFINE for a more detailed description.

Example:

```
MERGE          --colInMarker MarkerName
               --fileRef /test/referencefile.txt
                      --strMissing NA --strSep SPACE
               --colRefMarker rsID
               --strInSuffix .in
               --strRefSuffix .ref
               --blnInAll 1
               --blnRefAll 0
```

```
--blnWriteNotInRef 0
--blnWriteNotInIn 0
--strTag REF
```

Output:

The merged data set.

REPORT VARIABLES	*DESCRIPTIPON*
NotInRef	Number SNP from the input that are not available in fileRef.
NotInIn	Number SNP from fileRef that are not available in fileIn.

FILE OUTPUTS	*DESCRIPTION*
*.notinref.txt	If --blnWriteNotInRef 1, a separate file with SNPs from the input that are not available in fileRef.
*.notinin.txt	If --blnWriteNotInIn 1, a separate file with SNPs from fileRef that are not available in input.

MERGEEASYIN

Merges all defined input files. The defined <fileInTag> (see EASYIN command) will be added to the column names.

Input:

PARAMETER	*DESCRIPTION*
-- colInMarker	SNP column name of the input. Will be used for merging.
-- blnMergeAll	Boolean value to define inner/outer join. If set to 1 (TRUE), all SNPs from all input files will be present in the merged data-set. If set to 0 (FALSE), only intersecting SNPs from the input files will be present in the merged data-set. Optional. Default: 1 Please use: [0|1].

Example:

```
MERGEEASYIN          --colInMarker MarkerName
                     --blnMergeAll 0
```

Output:

The merged data set.

METAANALYSIS

Option 1: Conduct a fixed-effect inverse-variance weighted meta-analysis of N input strata using

$$\beta_{Overall} = \frac{\sum_{i=1}^{N} \beta_i w_i}{\sum_{i=1}^{N} w_i},\ SE_{Overall} = \sqrt{\frac{1}{\sum_{i=1}^{N} w_i}} \rightarrow \frac{\beta_{Overall}}{SE_{Overall}} \sim N(0,1) \rightarrow p_{Overall} \text{ with } w_i = 1/se_i^2$$

(Cox and Hinkley, 1979).

Option 2: Conduct a z-score based sample size weighted meta-analysis of M input strata using

$$z_{Overall} = \frac{\sum_{i=1}^{M} z_i \sqrt{N_i}}{\sqrt{\sum_{i=1}^{M} N_i}} \sim N(0,1) \rightarrow p_{Overall}$$

(Willer, et al., 2010).

Input:

PARAMETER	*DESCRIPTION*
--acolBETAs	Array of m effect-size, beta columns.
--acolSEs	Array of m standard error columns.
--acolZscores	Array of the z-score columns.
--acolNs	Array of the N, sample size columns.
--acolA1s	Array of m Allele 1 columns (Effect alleles).
--acolA2s	Array of m Allele 2 columns (Other alleles).
--colOutBeta	Name of the added pooled Beta column. Optional. Default: bmeta
--colOutSe	Name of the added pooled Standard Error column. Optional. Default: semeta
--colOutZscore	Name of the added pooled Z-score column. Optional. Default: zmeta
--colOutN	Name of the added total sample size column. Optional. Default: nmeta
--colOutP	Name of the added pooled P-Value column. Optional. Default: pmeta

IMPORTANT: Either define (--acolBETAs AND --acolSEs) to for Option 1
OR define (--acolZscores AND --acolNs) for Option 2.
The program will automatically recognize the formula to be applied.

Example:

```
METAANALYSIS    --acolBETAs beta.YOUNGMEN;beta.OLDMEN;beta.YOUNGWOMEN;beta.OLDWOMEN
                --acolSEs se.YOUNGMEN;se.OLDMEN;se.YOUNGWOMEN;se.OLDWOMEN
                --acolA1s A1.YOUNGMEN;A1.OLDMEN;A1.YOUNGWOMEN;A1.OLDWOMEN
                --acolA2s A2.YOUNGMEN;A2.OLDMEN;A2.YOUNGWOMEN;A2.OLDWOMEN
                --colOutBeta betaOverall
                --colOutSe seOverall
                --colOutP pOverall

METAANALYSIS    --acolZscores z.YOUNGMEN;z.OLDMEN;z.YOUNGWOMEN;z.OLDWOMEN
                --acolNs N.YOUNGMEN;N.OLDMEN;N.YOUNGWOMEN;N.OLDWOMEN
                --acolA1s A1.YOUNGMEN;A1.OLDMEN;A1.YOUNGWOMEN;A1.OLDWOMEN
```

```
--acolA2s A2.YOUNGMEN;A2.OLDMEN;A2.YOUNGWOMEN;A2.OLDWOMEN
--colOutZscore zOverall
--colOutN nOverall
--colOutP pOverall
```

Output:

Columns <colOutBeta>,<colOutSe> (or <colOutZscore>, <colOutN>) and <colOutP> will be added to the data-set. These columns contain the pooled (strata-combined) overall effect, standard error and P-Value respectively.

QQPLOT

Create QQ plot.

Input:

PARAMETER	*DESCRIPTION*
-- acolQQPlot	Array of P-Value columns that will be plotted into one graph. Please use: P-Value column names separated by ';'.
-- astrColour	Array of colours, used respectively for --acolQQPlot. Optional. Default: black. Please use: Any R-colour names or hexadecimal nomenclature (e.g. #FF0000 for red), separated by ';'
-- numPvalOffset	Numeric value. To increase the plotting speed, all SNPs with P-Values > numPvalOffset will be omitted from the plot. Optional. Default=1.
--blnYAxisBreak	Boolean value to define whether the y-Axis should be rescaled at the threshold --numYAxisBreak. If set, all plot values > numYAxisBreak will be linearly fitted into the upper 20% of the graphing area and all other values < numYAxisBreak will be linearly fitted into the lower 80% of the graphing area. Optional. Default: 0.
--numYAxisBreak	Numeric value at which the y-axis will be rescaled (if blnYAxisBreak=1). Optional. Default: 22
--blnLogPval	Boolean value to define whether the defined P-Value columns have already been (-log)-transformed. Optional. Default: 0 (expects P-Values in [0,1])
--blnPlotCI	Boolean value to define whether confidence bounds should be added to the plot. Please note that confidence bounds will be calculated for the first stated P-Value in acolQQPLot. Optional. Default: 0
--anumSymbol	Array of symbol integers, used respectively for --acolQQPlot. Optional. Default: 20. Please use: Any R-symbol integer (refers to the R plot parameter pch), separated by ';'
--anumCex	Array of symbol size magnification, used respectively for --acolQQPlot. Optional. Default: 1 Please use: Any R-symbol magnification (refers to the R plot parameter cex), separated by ';'
--blnCombined	Boolean value to define whether a combined line (QQPLOT of all acolQQPlot variables combined) should be added to the graph. Optional. Default: 0. Please use: [0|1].
--arcdExclude	Array of logical R-code criterions (separated by ;), used to remove SNPs from the plotting. Optional.
--arcdAdd2Plot	Array (separated by ';') of R - plotting functions (e.g. abline), that will be evaluated after the plot command. This can be used to add user-defined lines, points or text to the figure. Optional.
--fileRemove	If defined, an additional QQ-curve will be plotted only showing SNPs that lie >numRemovePosLim apart (in bp) from any region as defined in fileRemove. The fileRemove must contain columns 'Chr' and 'Pos' and the centered SNPs of the regions.
--numRemovePosLim	The position threshold in bp. Optional. Default: 500000
--strRemovedColour	Colour of the additional curve that excluded the loci defined in fileRemove. Optional. Default: green
--numRemovedSymbol	Symbol of the additional curve that excluded the loci defined in fileRemove. Optional. Default: 20

--numRemovedCex	Symbol size of the additional curve that excluded the loci defined in fileRemove. Optional. Default: 1
--colInChr	Column of the input that contains information about chromosome. If fileRemove is defined, this column is requested.
--colInPos	Column of the input that contains information about position. If fileRemove is defined, this column is requested.
Other graphic parameters:	
--strMode, --strFormat, --strAxes, --strXlab, --strYlab, --strTitle, --arcdAdd2Plot, --strPlotName, --numCexAxis --numCexLab, --numWidth, --numHeight, --anumParMar, --anumParMgp, --strParBty, --numParLas	Inherited SPLOT parameters. Can be used to influence graphical presentation. See SPLOT for a more detailed description.

Example:

```
QQPLOT --acolQQPlot Pmen;Pwomen
       --astrColour blue;red
       --anumSymbol 0;1
       --numPvalOffset 0.05
```

Output:

The input data set remains unchanged.

FILE OUTPUTS	*DESCRIPTION*
*.qq.[png|pdf]	QQ plot.

Example qq plot:

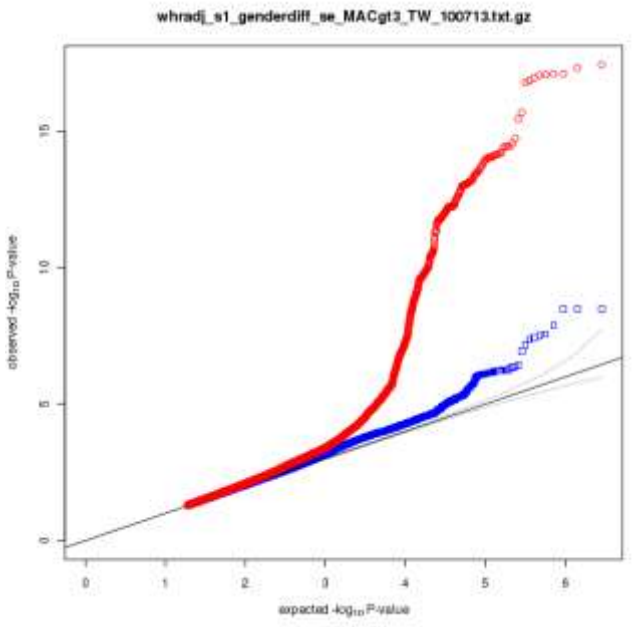


Figure. QQ plot showing women- and men-specific genome-wide association meta-analysis results for WHR_{adjBMI} (Randall, et al., 2013). Women results are colored in red, men in blue. The data shown is publically available from the GIANT consortium website www.broadinstitute.org/collaboration/giant.

REMOVECOL

Remove column.

Input:

PARAMETER	*DESCRIPTION*
--colRemove	Column that is supposed to be removed from the data set.

Example:

```
REMOVECOL     --colRemove P
```

Output:

The output data set will no more contain <colRemove>.

RENAMECOL

Rename column.

Input:

PARAMETER	*DESCRIPTION*
--colInRename	Old column name. If the specified colInRename is not available in the input, no action.
--colOutRename	New column name. If the specified colOutRename already exists in the data set, the suffix ".old" will be added to the existing column.

Example:

```
RENAMECOL     --colInRename Marker
              --colOutRename SNP
```

Output:

Column <colInRename> will be labelled <colOutRename> in the output data set.

RPLOT

Create a scatterplot of columns in the REPORT file.

Input:

PARAMETER	*DESCRIPTION*
-- rcdRPlotX	R-Code expression to define the x-values.
-- rcdRPlotY	R-Code expression to define the y-values.
Other graphic parameters: --strDefaultColour, --arcdColourCrit, --astrColour --numDefaultSymbol, --arcdSymbolCrit, --anumSymbol --numDefaultCex, --arcdCexCrit, --anumCex --strAxes, --strXlab, --strYlab, --strTitle, --arcdAdd2Plot --blnGrid, --strMode, --strPlotName, --strFormat --numCexAxis, --numCexLab, --numWidth, --numHeight --anumParMar, --anumParMgp, --strParBty	Inherited SPLOT parameters. Can be used to influence graphical presentation. See SPLOT for a more detailed description.

Example:

```
RPLOT  --rcdRPlotX maxSampleSize
       --rcdRPlotY GClambda
       --strAxes zeroequal
       --arcdAdd2Plot abline(h=1,col='red')
```

Output:

FILE OUTPUTS	*DESCRIPTION*
*.rplot.[png|pdf]	Report plot.

SPLOT

Create a scatterplot of columns in the GWA data-set.

Input:

PARAMETER	*DESCRIPTION*
--rcdSPlotX	R-Code expression to define the x-values.
--rcdSPlotY	R-Code expression to define the y-values.
Colouring parameters:	
--strDefaultColour	Default colour for the SNPs plotted. Any R-Colours are available. Also the hexadecimal nomenclature can be used: (e.g. #FF0000 for red). Optional. Default: black
--arcdColourCrit	Array of logical R-code criterions (separated by ;), used to distinguish colouring of the SNPs drawn. This overwrites the default colour for the SNPs that match the criterion.Optional.
--astrColour	Array of colours (separated by ;) to be used for arcdSPlotColourCrit respectively. Optional.
Symbol parameters:	
--numDefaultSymbol	Default R-Symbol for the SNPs plotted. Any R-Symbol integers are available (refers to the R plot parameter pch). Optional. Default: 20
--arcdSymbolCrit	Array of logical R-code criterions (separated by ;), used to distinguish symbols of the SNPs drawn. This overwrites the default symbol for the SNPs that match the criterion. Optional.
--anumSymbol	Array of symbols (separated by ;) to be used for arcdSPlotSymbolCrit respectively. Optional.
Symbolsize parameters:	
--numDefaultCex	Default Symbol size for the SNPs plotted. Any R Symbol size is available (refers to the R plot parameter cex). Optional. Default: 1
--arcdCexCrit	Array of logical R-code criterions (separated by ;), used to distinguish symbol sizes of the SNPs drawn. This overwrites the default symbol size for the SNPs that match the criterion.Optional.
--anumCex	Array of symbol sizes (separated by ;) to be used for arcdSPlotCexCrit respectively.Optional.
Confidence intervals:	
--blnPlotCI	Boolean value to define whether confidence bounds should be drawn to the points depicted. Default: 0
--rcdCIlengthX	R-Code expression to define the length of confidence interval (to either side) into x-direction. For example this could be set to 1.96*SE with SE being the standard error of a plotted BETA value on the x-axis.
--rcdCIlengthY	R-Code expression to define the length of confidence interval (to either side) into y-direction. For example this could be set to 1.96*SE with SE being the standard error of a plotted BETA value on the y-axis.
--strCIColour	Sting to indicate colour of the drawn confidence arrows. Default: 'grey'
General plotting parameters:	
--strAxes	X-/Y-axes alignment. Define plotting thresholds for x-axis [x0,x1] and y-axis [y0,y1]. Optional. Default: lim(NULL,NULL,NULL,NULL) Please use: [equal|4quad|4quadequal|zeroequal|lim(x0,x1,y0,y1)] (i) equal: Same x- and y-axis limits: x0=y0 and x1=y1 (ii) 4quad: All 4 quadrants will be drawn: x0=-x1 and y0=-y1 (iii) 4quadequal: All 4 quadrants will be drawn with the same x- and y-axis limits: x-axes limits == y-axes limits with lim=max(abs(x),abs(y)) (iv) zeroequal: x0=y0=0 and x1=y1=max(x,y) (v) lim(x0,x1,y0,y1): Define x- and y-axes limits (use NULL for default value)
--strXlab	X-axis label. Optional. Default: <rcdSPlotX>
--strYlab	Y-axis label. Optional. Default: <rcdSPlotY>
--strTitle	Plot title. Optional. Default: ''
--arcdAdd2Plot	Array (separated by ';') of R - plotting functions that will be evaluated after the plot command. This can be used to add user-defined lines, points or text to the figure.
--blnGrid	Boolean value to define whether grid lines should be drawn on the plot. Optional. Default: 1
--strMode	Set the plotting mode. A singleplot means that one image file will be created for each input. A subplot means that all graphs will be combined to a single png file (arranging the single graphs in rows/columns). Optional. Default: singleplot. Please use: [singleplot|subplot]
--strPlotName	String to define a name for the plot. This will be added to the output file name thus can be used to distinguish images in the output folder. Optional. Default: sp
--strFormat	Set the image file format, pdf or png. Optional. Default: png; Please use: [png|pdf]
--numCexAxis	Size of the axis, refers to R plot parameter cex.axis. Optional. Default: 1
--numCexLab	Size of the axis labels, refers to R plot parameter cex.lab. Optional. Default: 1
--numWidth	Width of the plot in pixel. Optional. Default: 640 (6 for pdf)
--numHeight	Height of the plot in pixel. Optional. Default: 640 (6 for pdf)
--anumParMar	Numerical vector of the form 'c(bottom, left, top, right)' which gives the number of lines of margin to be specified on the four sides of the plot. This refers to the R plot parameter 'mar'. Optional. Default: 'c(5, 4, 4, 2) +0.1'
--anumParMgp	The margin line (in 'mex' units) for the axis title, axis labels and axis line. Note that 'mgp[1]' affects 'title'

	whereas 'mgp[2:3]' affect 'axis'. This refers to the R plot parameter 'mgp'. Optional. Default: 'c(3, 1, 0) '
--strParBty	A character string which determined the type of 'box' which is drawn about plots. If 'bty' is one of '"o"' (the default), '"l"', '"7"', '"c"', '"u"', or '"]"' the resulting box resembles the corresponding upper case letter. A value of '"n"' suppresses the box. This refers to the R plot parameter 'bty'. Optional. Default: 'o'
--numParLas	Numerical value indicating the alignment of axis labels. This refers to the R plot parameter 'las'. Optional. Default: 0 (always parallel to the axis)

Example:

```
SPLOT  --rcdSPlotX Freq
       --rcdSPlotY -log10(Pvalue)
       --strDefaultColour black
       --numDefaultSymbol 20
       --arcdColourCrit N<110000;N<80000
       --astrColour purple;red
       --arcdSymbolCrit (Freq<=0.05|Freq>=0.95);(Freq<=0.01|Freq>=0.99)
       --anumSymbol 0;1
       --strAxes lim(0,NULL,0,NULL)
       --strTitle PoverFreq
       --arcdAdd2Plot abline(v=0.05);abline(v=0.95);abline(h=-log10(5e-8))
```

Output:

The input data set remains unchanged.

FILE OUTPUTS	*DESCRIPTION*
*.sp.[png|pdf]	Report plot.

Example scatter plot:

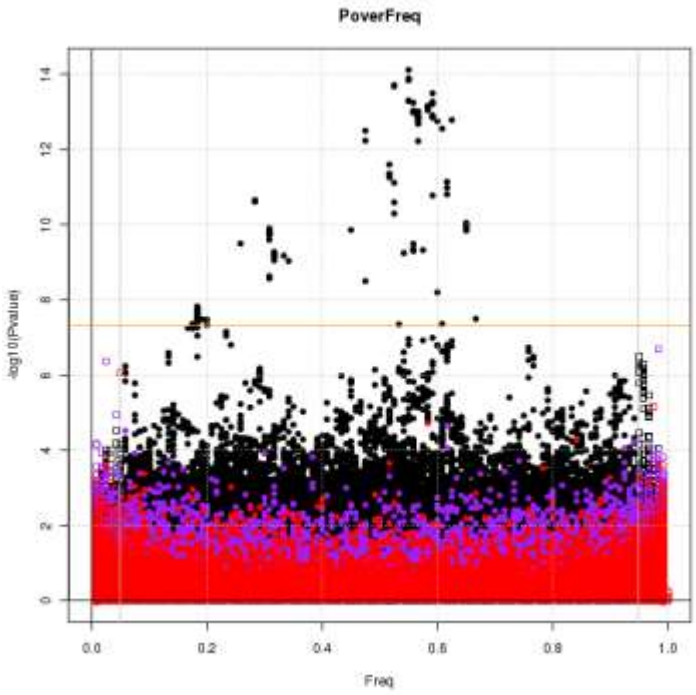


Figure. Scatterplot contrasting association P-Values (on –log10 scale) versus allele frequency, color coded by the sample size (purple < 70K, red < 50K), of the respective SNP association test. Square symbols indicate minor allele frequencies <0.05 and unfilled circles indicate minor allele frequencies <0.01. The data shown is publically available data for WHR_{adjBMI} (Heid, et al., 2010) , available from the GIANT consortium website www.broadinstitute.org/collaboration/giant.

STRSPLITCOL

Splits each column entry according to a defined character string and creates a new column containing only the i th result of the string spit.

Input:

PARAMETER	*DESCRIPTION*
--colSplit	Column for which the string splitting will be performed.
--strSplit	Character string according to which, each entry of colSplit will be splited (compare R-function strsplit).
--numSplitIdx	Integer value to specify which part of each output should be taken forward to the new array.
--colOut	Name of the new output column.

Say column colSplit equals array c("chr1_111", "chr2_222"):

- With "--strSplit _" and "--numSplitIdx 1", the new column c("chr1","chr2") will be created
- With "--strSplit _" and "--numSplitIdx 2", the new column c("111","222") will be created

Example:

```
STRSPLITCOL         --colSplit ChrPosId
                    --strSplit _
                    --numSplitIdx 1
                    --colOut Chromosome
```

Output:

The output data set will contain a new column <colOut> that carries the extracted information.

WRITE

Write data set. Output will be named
/pathOut/[strWritePrefix]fileInShortName[strWriteSuffix].[txt|gz]

Input:

PARAMETER	*DESCRIPTION*
-- strMode	Mode to write the current data set. Either as plain text- or as compressed gzipped-file. Optional. Default: txt Please use: [txt|gz]
-- strPrefix	Set file prefix. Optional. Default: ''
-- strSuffix	Set file suffix. Optional. Default: ''
-- strSep	Set file separator. Optional. Default: TAB Please use: [TAB,SPACE,COMMA]
-- strMissing	Set missing character. Default: NA. Optional. Default: NA

Example:

```
WRITE  --strMode txt
       --strPrefix CLEANED.
       --strSuffix .TW
       --strSep TAB
       --strMissing .
```

Output:

REPORT VARIABLES	*DESCRIPTIPON*
numSNPsOut	Number of SNPs in the last written file.

FILE OUTPUTS	*DESCRIPTION*
strPrefix.*<fileIn>*.strSuffix.[gz|txt]	Output file.

QC FUNCTIONS

AFCHECK

Check allele frequencies and strand orientation by creating an allele frequency scatter plot of each input file versus a reference.

Input:

PARAMETER	*DESCRIPTION*
--colRefFreq	Column name of the reference allele-frequency.
--colInFreq	Column name of the input allele-frequency. In case the allele direction will be switched in order to match the reference alleles, this column will be adjusted for the respective SNPs by (1-colInFreq).
--numLimOutlier	Allele frequency difference threshold that will be used to define outliers. The number of outlying SNPs will be written to the REPORT and the outlying SNPs itself will be written to a separate file in the output directory. Optional. Default: 0.2
--blnWriteOutlier	Boolean value to indicate whether outlying SNPs that differ > numLimOutlier in terms of allele frequency from the reference frequency, should be written to a separate file in the output path. Optional. Default: 1. Please use: [0|1].
--blnRemoveOutlier	Boolean value to indicate whether outlying SNPs that differ > numLimOutlier in terms of allele frequency from the reference frequency, should be excluded from the input. Optional. Default: 0. Please use: [0|1].
--blnPlotAll	Boolean value to indicate whether all SNPs should be plotted. If set to 0, ONLY outlying SNPs that differ > numLimOutlier in terms of allele frequency from the reference frequency, will be plotted. Optional. Default: 0 (only outlier are drawn). Please use: [0|1].
--blnRemoveMismatch 1, --blnRemoveInvalid 1, --blnRemoveRefInvalid 1 --blnWriteMismatch 1 --blnWriteInvalid 1 --colRefA1, --colRefA2, --colInA1 , --colInA2, --colRefStrand, --colInStrand, --blnMetalUseStrand,	Inherited ADJUSTALLELES parameters. See ADJUSTALLELES for a more detailed description. Parameter values are given if they differ from the default ADJUSTALLELES values.
--fileRef, --strRefSuffix .ref, --strInSuffix, --colRefMarker, --colInMarker, --blnRefAll, --blnInAll, --blnWriteNotInRef, --blnWriteNotInIn	Inherited from MERGE. See MERGE for a more detailed description. Parameter values are given if they differ from the default MERGE values.
--strMissing , --strSeparator, --acolIn, --acolInClasses, --acolNewName	Inherited from DEFINE. Can be used for --fileRef. See DEFINE for a more detailed description.
--strDefaultColour blue, --numDefaultCex 0.1, --arcdColourCrit abs(ref-in)>LimOutlier, --astrColour red, --strAxes lim(0,1,0,1), --arcdAdd2Plot abline(a=0,b=1,col='red',lty=1), --numDefaultSymbol, --strXlab, strYlab, --numCexAxis, --numCexLab, --numWidth,	Inherited from SPLOT. Can be used to influence graphical presentation. See SPLOT for a more detailed description. Parameter values are given if they differ from the default SPLOT values.

--numHeight, --anumParMar, --anumParMgp, -
-strParBty, --strFormat, --blnGrid

Inherited parameters can be used with AFCHECK directly.

In particular, setting --fileRef at AFCHECK can be helpful if the input does not (yet) contain reference alleles and reference allele frequencies and if reference data needs to be merged from external data (this may replace an extra MERGE and ADJUSTALLELES step in the ecf-file).

If the input already contains reference alleles and reference allele frequencies there is no need to use --fileRef or any other inherited parameters.

The defined input alleles, frequency and betas will be aligned to the defined reference alleles.

Example:

```
AFCHECK      --colInMarker ChrPosID
             --colInStrand Strand
             --colInA1 Effect_allele
             --colInA2 Other_allele
             --colInFreq EAF
             --fileRef /path2reffiles/AlleleFreq_HapMap_CEU.v1.txt.gz
                    --acolIn ChrPosID;A1;A2;Freq1
                    --acolInClasses character;character;character;numeric
             --colRefMarker ChrPosID
             --colRefA1 A1
             --colRefA2 A2
             --colRefFreq Freq1
             --blnMetalUseStrand 1
```

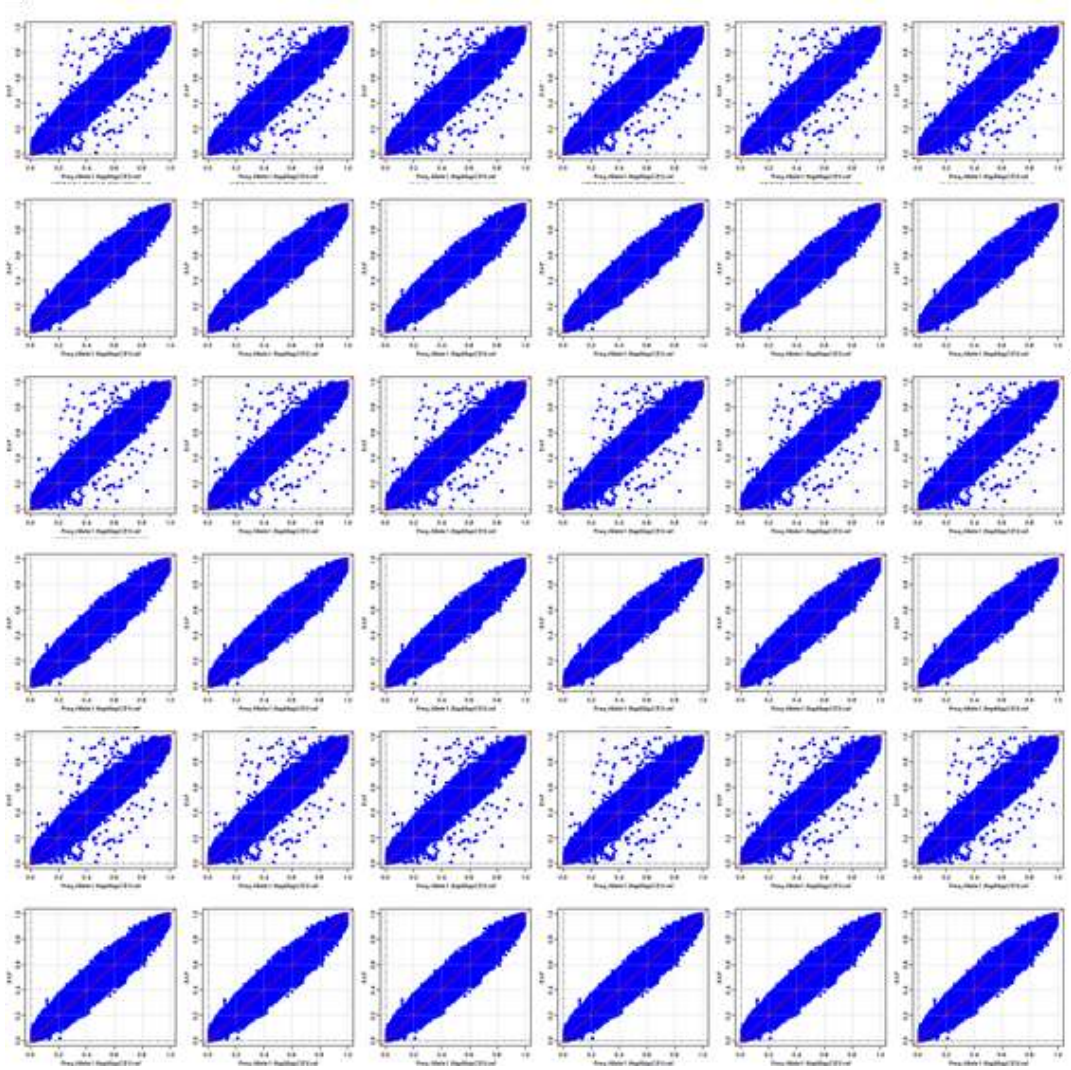
Output:

REPORT VARIABLES	*DESCRIPTIPON*
cor_<colRefFreq>_<colInFreq>	Pearson Correlation between the input and the reference allele frequency after aligning all directions to the reference.
numOutlier	Number of outlying SNPs that differ > numLimOutlier (0.2 by default) in terms of allele frequency from the reference allele frequency.
Checked, StrandChange, AlleleMatch, AlleleChange, n4AlleleMatch, n4AlleleChange, AlleleMismatch, AlleleInMissing, AlleleInInvalid, StrandInInvalid, AlleleRefMissing, AlleleRefInvalid, StrandRefInvalid	Inherited ADJUSTALLELES variables. Only present if --colInA1, --colInA2, --colRefA1 and --colRefA2 are used. Please see ADJUSTALLELES for a more detailed description.
NotInRef, NotInIn	Inherited MERGE variables. Only present if --fileRef is used. Please see MERGE for a more detailed description.

FILE OUTPUTS	*DESCRIPTION*
*.AFCHECK.[png|pdf]	Panel of allele frequency plots.
*.outlier.txt	If --blnWriteOutlier 1, a separate file will be written to pathOut that contains the outlying SNPs.

*.mismatch.txt, *. invalid.txt	Inherited ADJUSTALLELES output. Only present if --colInA1, --colInA2, --colRefA1 and --colRefA2 are used. Please see ADJUSTALLELES for a more detailed description.
*.notinref.txt, *.notinin.txt	Inherited MERGE output. Only present if --fileRef is used. Please see MERGE for a more detailed description.

Example AFCHECK plots panel:



CLEANDUPLICATES

Clean duplicates. Number of duplicates will be written to the report into variable numDuplicates.

Inputs:

PARAMETER	*DESCRIPTION*
-- colInMarker	Column name to check for duplicates.
-- strMode	Set mode for handling of the duplicates. Optional. Default: keepfirst Please use: 'keepfirst' to keep the first and to exclude all latter appearances of a SNP; 'removeall' to remove all duplicated SNPs; 'keepall' to keep all duplicated SNPs and to not remove any rows; 'samplesize' to keep the duplicated SNP with the highest sample size;
-- colN	If strMode is set to "samplesize",the column name of the samplesize column needs to be defined here.

Example:

```
CLEANDUPLICATES     --strMode samplesize
                    --colN N
```

REPORT VARIABLES	*DESCRIPTIPON*
numDuplicates.<colInMarker>	Number of duplicated SNPs in defined column colInMarker.

FILE OUTPUTS	*DESCRIPTION*
*.duplicates.<colInMarker>.txt	File containing the detected duplicated SNPs.

CREATECPTID

Creates a unique marker identifier column called 'cptid' (chromosome-position-type-id) that uses the format according to '<CHR>:<POSITION><TYPE>' with TYPE being ':ID' for INDELs or blank '' for SNPs.

Examples: SNP rs181588098 will be named 15:84093287
INDEL 1:739141:AT_A (IMPUTE format) will be named 1:739141:ID.
INDEL chr1:739141:D (MACH format) will be named 1:739141:ID.

Input:

PARAMETER	*DESCRIPTION*
--colInMarker	Column name of the input marker (required).
--colInA1	Column name of the input Allele1 (required).
--colInA2	Column name of the input Allele2 (required).
--colInChr	Column name of the input chromosome (required if column should be used to obtain the cptid).
--colInPos	Column name of the input position (required if column should be used to obtain the cptid).
--fileMap	Path to the mapping file (required if mapping file should be used to obtain the cptid).
--colMapMarker	Column name of the mapping file marker (optional, default: 'rsmid').
--colMapChr	Column name of the mapping file chromosome (optional, default: 'chr').
--colMapPos	Column name of the mapping file position (optional, default: 'pos').
--blnUseInMarker	Boolean value to indicate whether CHR and POS should be extracted from the given input marker column by reformatting (optional, default: 1).

The function will obtain TYPE from the allele columns of the input, i.e., if alleles are coded as I/D the function will set the TYPE to ':ID' (INDEL), otherwise to SNP.

The function provides three ways to obtain CHR and POS (used for the 'cptid') that depend on the given parameters and the columns in the data set:

1) Using a mapping file that contains a marker column that uses the same marker format as the input (e.g., rs-IDs) and columns with the respective chromosome and position information (parameters --fileMap, --colMapMarker, --colMapChr, --colMapPos) . The function will obtain CHR and POS from the mapping file.
2) Using the input marker column (--blnUseInMarker 1) that may allow to extract CHR and POS by reformatting the marker name:
   Examples: MarkerName 'chr1:123:AT_T' → cptid = 1:123:ID
   MarkerName 'chr1:123' → cptid = 1:123
   MarkerName '1:123:AT_T' → cptid = 1:123
   MarkerName 'c1b123INDEL' → cptid = 1:123
3) Using given columns for chromosomal and position information (--colInChr, --colInPos)

If all three options are to be used together, the function will first obtain the cptid from the mapping file, then from the given marker column and at last use given chromosomal and position information.

If none of the three ways can successfully obtain the cptid, the original marker name will be copied into the cptid from the given --colInMarker column and these mismatches will be written to a separate file in the output.

Example:

```
CREATECPTID  --colInMarker rsID
             --colInA1 EFFECT_ALLELE
             --colInA2 NON_EFFECT_ALLELE
             --colInChr CHR
             --colInPos POS
      --fileMap /path2Map/rsmid_map.1000G_ALL_p1v3.merged_mach_impute.v1.txt.gz
             --colMapMarker rsmid
             --colMapChr chr
             --colMapPos pos
# This example makes use of all three ways to obtain the cptid.
```

Output:

The output data set will only contain an additional column called 'cptid'.

REPORT VARIABLES	*DESCRIPTIPON*
CPTID.frommap	If--fileMap is set, this is the number of cptid's that were created based on CHR and POS from the mapping file.
CPTID.from_chrpos_format	If--blnUseInMarker is set to 1 (default), this is the number of cptid's that were created from a "[chr]<CHR>:<POS>[:*]" format, e.g., '1:123:ID' from from 'chr1:123:AT_T'.
CPTID.from_cb_format	If--blnUseInMarker is set to 1 (default), this is the number of cptid's that were created from a "c<CHR>b<POS>[INDEL|SNP]" format, e.g., '1:123:ID' from from 'c1b123INDEL'.
CPTID.from_chr_pos	If --colInChr and –colInPos are set, this is the number of cptid's that were created based on CHR and POS from the input data set.
CPTID.nomatch	Number of SNPs for which a 'cptid' could not be successfully created. The original names of the SNPs are copied into the 'cptid' column.

FILE OUTPUTS	*DESCRIPTION*
*.CPTID.nomatch.txt	This file contains the SNPs for which a 'cptid' could not be successfully created. The original names of the SNPs are copied into the 'cptid' column.

FLIPSTRAND

Flip strand for all SNPs that are present in a reference file. Here flipping strand simply means to change “+” to “-” or vice versa. Neither alleles nor frequencies or effect estimates will be changed.

Input:

PARAMETER	*DESCRIPTIPON*
--colInMarker	Marker column name of the input file.
--colInStrand	Strand column name of the input file.
--fileRef	Path to the reference file. SNPs listed in this file will be used for flipping strand (from + to – or vice versa).
--colRefMarker	Marker column name of the reference file.

Example:

```
FLIPSTRAND    --colInMarker MarkerName
              --colInStrand Strand
              --fileRef /path2ref/refssnps.txt
              --colRefMarker SNP
```

The Strand value of the SNPs defined in <fileRef> will be flipped (+ to – or vice versa).

Output:

REPORT VARIABLES	*DESCRIPTIPON*
numFlipStrand	Number of SNPs of which the strand has been flipped.

HARMONIZEALLELES

Harmonizes alleles so that all output alleles match 'A','C','G' or 'T' for SNPs or 'I' (Insertion), 'D' (Deletion) for INDELs. To accomplish this, the function

i) removes SNPs that are missing both alleles (A1 = NA and A2 = NA)

ii) reformats different deletion codes to 'D' and sets the other to 'I'
Examples:
- A1=NA, A2='G' → A1='D', A2='I'
- A1='<DEL>', A2='G' → A1='D', A2='I' (<DEL> may be from MACH reference)
- A1='-' & A2='G' → A1='D', A2='I' ('-' may be from IMPUTE reference)

iii) reformats MACH's R/I and R/D coding to D/I (to only have two characters for INDELs)
Examples:
- A1='R', A2='I' → A1='D', A2='I'
- A1='R', A2='D' → A1='I', A2='D'

iv) reformats sequence coding to D/I
Example: A1=' GCAT', A2=' GCT' → A1='I', A2='D'

v) removes all SNPs that do not match 'A','C','G' or 'T' and all INDELs that do not match 'I', 'D'.

Input:

PARAMETER	*DESCRIPTION*
--colInA1	Column name of the input Allele1.
--colInA2	Column name of the input Allele2.

Example:

```
HARMONIZEALLELES    --colInA1 EFFECT_ALLELE
                    --colInA2 OTHER_ALLELE
```

Output:

The output data set will only contain allele coded 'A','C','G' or 'T' for SNPs and 'I', 'D' for INDELs.

REPORT VARIABLES	*DESCRIPTIPON*
HA.numDrop_BothAllelesMissing	Number of SNPs that were removed due to i)
HA.num_Recoded_DEL	Number of SNPs that were reformatted according to ii)
HA.num_Recoded_MACH_R	Number of SNPs that were reformatted according to iii)
HA.num_Recoded_SEQ	Number of SNPs that were reformatted according to iv)
HA.numDrop_InvalidAlleles	Number of SNPs that were removed due to v)

FILE OUTPUTS	*DESCRIPTION*
*. HA.numDrop_InvalidAlleles.txt	This file contains the SNPs/INDELs that were removed from the data set due to v), i.e. SNPs/INDELs for which the alleles could not be recoded to A,C,G,T or I,D.

PZPLOT

Check if P-Value gathered from z-statistic beta/se matches the P-Value stated in the file by creating a scatter plot of the P-Values for each input file. To increase plotting speed, only nominally significant SNPs P<0.05 will be drawn.

Input:

PARAMETER	*DESCRIPTION*
--colBeta	Column name of the betas.
--colSe	Column name of the SEs.
--colPval	Column name of the P-Values.
--numPvalOffset	Numeric value. To increase the plotting speed, all SNPs with P-Values > numPvalOffset will be omitted from the plot. Optional. Default: 0.05

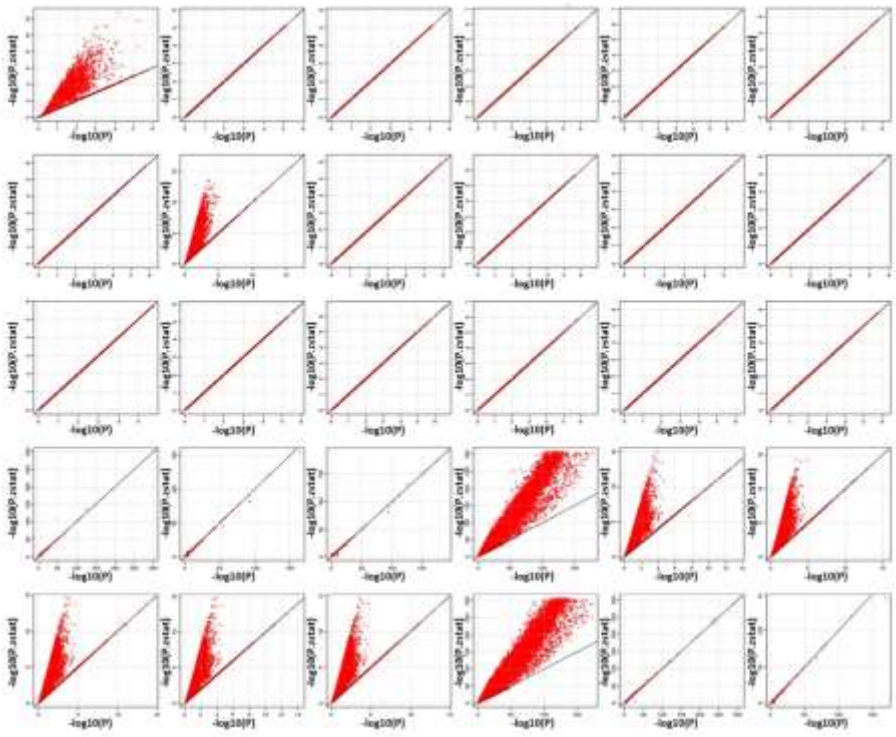
Example:

```
PZPLOT --colBeta BETA
       --colSe SE
       --colPval P
```

Output:

FILE OUTPUTS	*DESCRIPTION*
*.PZ-PLOTS.[png|pdf]	Panel of PZ-plots.

Example PZ-plots panel:



RENAMEMARKER

Rename SNP names using a reference data set that solely contains two columns, one with old marker names, one defining the respective new marker names. The output column will be named by '--colRenameNewMarker'.

Input:

PARAMETER	*DESCRIPTION*
--colInMarker	Marker column name of the input GWA data-sets, of which the snp names will be renamed.
--fileRename	File containing two columns: A column with old marker names and a column with respective new marker names.
--colRenameOldMarker	Column in fileRename containing the old marker names.
--colRenameNewMarker	Column in fileRename containing the new marker names.
--blnWriteRenamed	Boolean value to define whether the renamed SNPs should be written to a separate file in the output. Optional. Default: 0 Please use: [0|1]
--blnWriteNomatches	Boolean value to define whether the nomatching SNPs (SNPs from the input that are not listed in colRenameOldMarker of the reference) should be written to a separate file in the output. Optional. Default: 0 Please use: [0|1]
--blnSaveOldMarker	Boolean value to define whether the old marker column should be saved in column '<colInMarker>.old'. Optional. Default: 1 (Old Marker column will be saved to column '<colInMarker>.old') Please use: [0|1]
--blnRetainNoMatches	Boolean value to define whether no matches (a SNP in colInMarker that is neither contained in colRenameOldMarker nor in colRenameNewMarker) should be retained in the output using the existing value. If set to 0 a no matching SNP will be set to NA. Optional. Default: 1 Please use: [0|1]

Example:

```
RENAMEMARKER  --colInMarker MarkerName
              --fileRename /file2ref/FileThatWillBeUsedForRenaming.txt
              --colRenameOldMarker rsID
              --colRenameNewMarker ChrPosID
              --blnWriteRenamed 0
              --blnWriteNomatches 0
              --blnSaveOldMarker 1
              --blnRetainNoMatches 1
```

Output:

The output data set will carry renamed markers in column

REPORT VARIABLES	*DESCRIPTION*
<colRenameNewMarker> and carry the old marker column in column '<colInMarker>.old' (if blnSaveOldMarker=1). numRenamedMatch	Number matching SNPs, i.e. SNPs that either already carried the new marker name or SNPs that were renamed using fileRename.

FILE OUTPUTS	*DESCRIPTION*
*.renamed.marker.txt	If --blnWriteRenamed 1, a separate file containing all SNPs that were renamed.
*.nomatch.marker.txt	If --blnWriteNomatches 1, a separate file with SNPs that did not match the old or new marker column of fileRename.

REFERENCES

Cox, D.R. and Hinkley, D.V. (1979) *Theoretical statistics*. Chapman and Hall ;
distributed in U.S. by Halsted Press, London
New York.

Heid, I.M., *et al.* (2010) Meta-analysis identifies 13 new loci associated with waist-hip ratio and reveals sexual dimorphism in the genetic basis of fat distribution, *Nature genetics*, **42**, 949-960.

Randall, J.C., *et al.* (2013) Sex-stratified genome-wide association studies including 270,000 individuals show sexual dimorphism in genetic loci for anthropometric traits, *PLoS genetics*, **9**, e1003500.

Willer, C.J., Li, Y. and Abecasis, G.R. (2010) METAL: fast and efficient meta-analysis of genomewide association scans, *Bioinformatics*, **26**, 2190-2191.