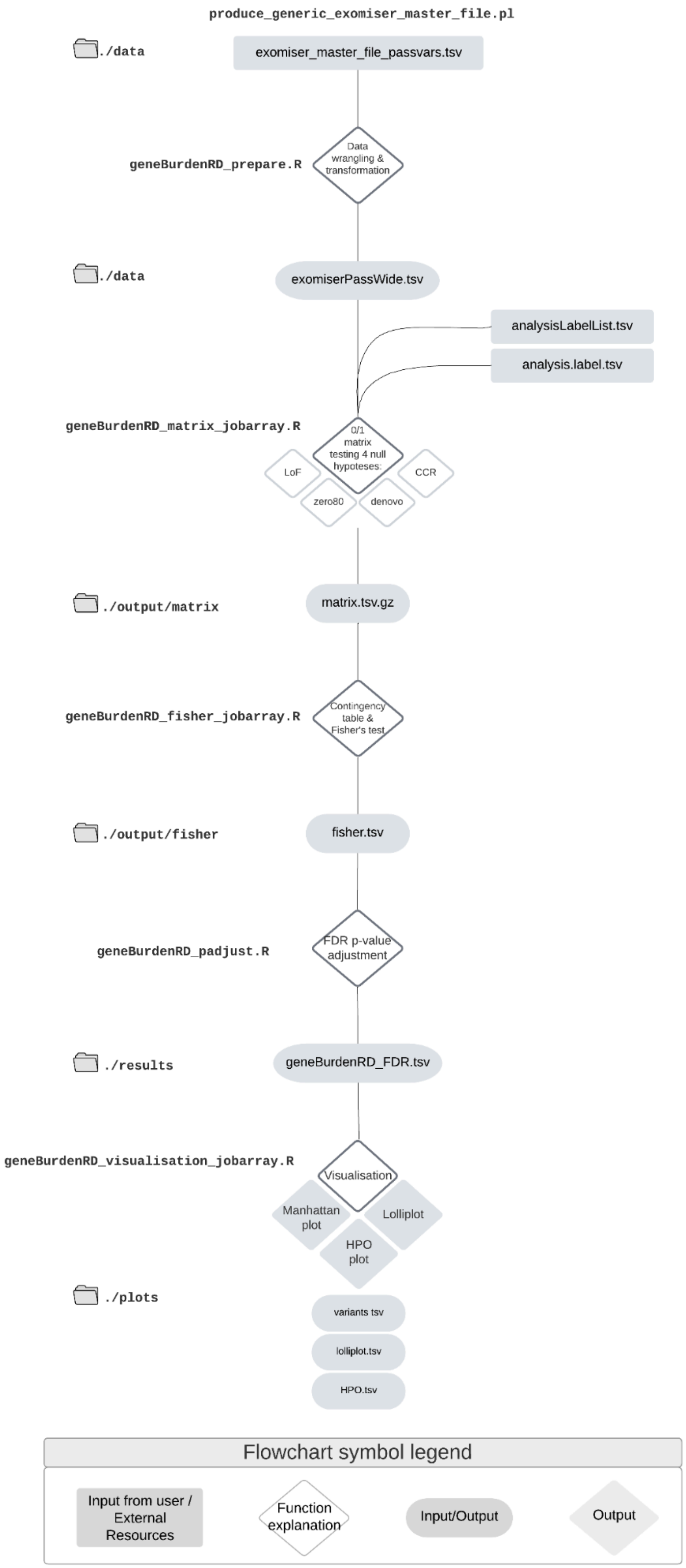
Overview

`geneBurdenRD` is an open-source R framework that allows users to perform gene burden testing of variants in user-defined cases versus controls from rare disease sequencing cohorts. The input to the framework is a file obtained from processing Exomiser output files for each of the cohort samples, a file containing a label for each case-control association analysis to perform within the cohort and a (set of) corresponding file(s) with user-defined identifiers and case/control assignment per each sample. Cases and controls in a cohort could be defined in many ways, for example, by recruited disease category as we have done for the 100KGP analysis below, by specific phenotypic annotations or phenotypic clustering. The framework will then assess false discovery rate (FDR)-adjusted disease-gene associations where genes are tested for an enrichment in cases vs controls of rare, protein-coding, segregating variants that are either (i) predicted loss-of-function (LoF), (ii) highly predicted pathogenic (Exomiser variant score >= 0.8), (iii) highly predicted pathogenic and present in a constrained coding region (CCR; 13) or (iv) de novo (restricted to only trios or larger families where de novo calling was possible and provided by the user). As well as various output files annotating these case-control association tests, Manhattan and volcano plots are generated summarising the FDR-adjusted p-values of all the gene-based tests for each case-control association analysis, along with lollipop plots of the relevant variants in cases and controls and plots of the hierarchical distribution of the Human Phenotype Ontology (HPO) case annotations for individual disease-gene associations.


produce_generic_exomiser_master_file.pl
./data
exomiser_master_file_passvars.tsv
geneBurdenRD_prepare.R
Data wrangling & transformation
./data
exomiserPassWide.tsv
analysisLabelList.tsv
analysis.label.tsv
geneBurdenRD_matrix_jobarray.R
0/1 matrix testing 4 null hypoteses:
LoF
zero80
denovo
CCR
./output/matrix
matrix.tsv.gz
geneBurdenRD_fisher_jobarray.R
Contingency table & Fisher's test
./output/fisher
fisher.tsv
geneBurdenRD_padjust.R
FDR p-value adjustment
./results
geneBurdenRD_FDR.tsv
geneBurdenRD_visualisation_jobarray.R
Visualisation
Manhattan plot
Lolliplot
HPO plot
./plots
variants tsv
lolliplot.tsv
HPO.tsv
Flowchart symbol legend
Input from user / External Resources
Function explanation
Input/Output
Output

System requirements

- **All software dependencies and operating systems (including version numbers)**

The R framework only demands the presence of R or Rstudio and a standard computer with sufficient RAM to accommodate in-memory operations. The following R packages will need to be installed (if not already) via install.packages('packageName') from R:

```
library("tidyverse")
library("data.table")
library("reshape2")
library("biomaRt")
library("ggplot2")
library("ggrepel")
library("httr")
library("drawProteins")
library("ensembldb")
library("AnnotationHub")
library("ontologyIndex")
library("ontologyPlot")
```

- **Versions the software has been tested on**

The R framework was tested using R/4.1.0.

- **Any required non-standard hardware**

Non-standard hardware is not required but to analyse larger cohorts it is useful to have access to a large, high compute cluster to run as many jobs in parallel as possible and reduce the overall run-time.

Installation guide

Installation is from our GitHub repository, e.g. git clone https://github.com/whri-phenogenomics/geneBurdenRD.git or a download from https://github.com/whri-phenogenomics/geneBurdenRD/tree/master#:~:text=Download-,ZIP

As the installation involves simply cloning/downloading from a GitHub repo it only takes a matter of seconds on any computer.

Demo

The installation comes with the following example data:

1. example **exomiser_master_file_passvars.tsv** file that can be used as input to the main R analysis. This file contains processed Exomiser output (using produce_generic_exomiser_master_file_final.pl) from 10 singleton cases with several hundred rare, coding variants per case that passed the Exomiser filters including a heterozygous, pathogenic, FGFR2:ENST00000358487.10:c.1694A>C:p.(Glu565Ala) variant that causes Pfeiffer syndrome and the annotated Brachydactyly, Craniosynostosis, Broad thumb, Broad hallux HPO terms along with processed output from 100 singleton controls that do not have Pfeiffer syndrome or this *FGFR2* variant.
2. **analysisLabelList.tsv** contains a header row (analysis.label, analysis), followed by rows that specify a code representing the tested disease (e.g., PFFS) and an explanation for the disease tested (e.g., Pfeiffer syndrome).
3. **PFFS.tsv** contains a header row (sample.id, caco,caco.denovo) followed by 10 rows for case_1 to case_10 with a 1 in the caco column to signify these are the cases and then 100 rows for control_1 to control_10 with a 0 in the caco column to indicate controls.

How to run the analysis on the demo data

```
-  git clone or download zip geneBurdenRD as described above
-  sh geneBurdenRD_prepare.sh
-  sh geneBurdenRD_matrix_jobarray.sh
-  sh geneBurdenRD_fisher_jobarray.sh
-  sh geneBurdenRD_padjust.sh
-  sh geneBurdenRD_visualisation_jobarray.sh
```

Expected output:

The `./results` folder includes the geneBurdenRD FDR tsv file which provides a summary of statistics for all signals and includes:

- analysis.label
- analysis
- gene	Exomiser gene
- test	null hypothesis tested (LoF, zero80, denovo or CCR)
- pvalue		p-value after one-sided Fisher Exact test
- p.adjust.fdr	p-value after false discovery rate (FDR) adjustment
- or	odds ratio
- d	number of cases with the event
- totcases		total number of cases (b+d)
  totcontrols	total number of controls (a+c)
- totgenestested		total number of genes tested
- tottests		total number of tests

- lowclor odds ratio lower confidence limit
- upclor odds ratio upper confidence limit
- a number of controls without the event
- b number of cases without the event
- c number of controls with the event
- filename path to corresponding fisher.tsv file
- bonferroni.cutoff

The output of the visualization script is explained below.

The `./plots` folder includes individual subfolders for each analysis, each containing at least one significant signal.

Each analysis subfolder contains:

► one `Manhattan-like plot` for the analysis: the x-axis is the chromosome position and the y-axis is the -log10 of the p-value after FDR adjustment. Each dot represents a test that passed the minimum requirements for the contingency table before running the Fisher exact test (d≥1, TotCases≥5 and TotEvent≥4). Each colour represents the null hypothesis that was tested. The dashed line represents the user-selected significant thresholds, indicating a p-value considered significant after adjusting for false discovery rate (FDR) that is lower than the user-specified threshold.

► one `volcano plot` for the specific disease: the x-axis is the log2 of the odds ratio and the y-axis is the -log10 of the p-value after FDR adjustment. Each dot represents a test that passed the minimum requirements for the contingency table before running the Fisher exact test (d≥1, TotCases≥5 and TotEvent≥4). Each colour represents the null hypothesis that was tested. The dashed lines represent the user-selected significant thresholds, indicating a p-value considered significant after adjusting for false discovery rate (FDR) that is lower than the user-specified threshold and odds ratio >= 3.

► one subfolder per each significant gene-test: * please note that the null hypothesis tested are LoF, zero80 (Exomiser variant score>=0.8), denovo or CCR (constrained coding regions).

Each gene-test subfolder contains:

● one `lolliplot` for each significant gene-test: the lolliplot shows the variants in the gene found in cases contributing to the gene burden signal for the specific test (in grey). Please note that lollies coloured in yellow show variants passing the gene-test that were eventually excluded from the analysis (caco == NA). The x-axis represents the amino acid chain and its annotated protein domain (Uniprot). Each lolly indicates a variant by its protein change annotated on one single transcript specified in the plot, the frequency of the variant in the contributing cases is shown on the y-axis. Its shape indicates the genotype found in the proband. The colour indicates the type of variant and the variant’s functional annotation. If the variants have both a p. change annotation and a number in parenthesis ( ) means that the original p. change was annotated on a different transcript and the amino acid position in parenthesis indicates the re-annotation on the selected transcript, if the only annotation available indicates a number in parenthesis ( ) it means that the variants were in the non-coding or splice region for that transcript, therefore the lolly was placed on the closest predicted amino acid. * Please be aware that the re-annotated non-coding or splice region variants provided are estimations of the nearest amino acid and should be verified for accuracy. For comprehensive details on all variants depicted in the lolliplot, refer to the lolliplot tsv table to retrieve the original information.

- one `lolliplot tsv` table: it contains one row per each variant in the lolliplot found in cases and excluded probands and gives additional information about each variant. It includes all the columns described below for the tsv variant file * compared to the tsv variant file the lolliplot tsv file has some additional columns listed below :

`Patient_ID` fake patient ID that can be used to interpret the “hpo_plot_freq.jpg”

`gene.symbol`, `hgvs_stranscript`, `hgvs_c_change` and `hgvs_p_change`
HGVS gene symbol, HGVS transcript, HGVS c. change and HGVS p. change

`genotype` genotype shown in the lolliplot (hom=homozygous or hemizygous, het=heterozygous, comp_het= compound heterozygous)

`select_transcript` transcript selected for the lolliplot

`protein.change` protein change annotated on the selected transcript

`fixed` Y or N if the variant needed to be re-annotated to the selected transcript

`var.aanum` amino acid number on lolliplot

For better visualization clarity, please be aware that the limit for variants displayed in the lolliplots has been set to 100.

- one `HPO plot` showing all the HPO terms and their common ancestor terms found in cases using the fake patient ID in the lolliplot tsv table. The colour of the HPO node indicates the frequency of their shared clinical features. * if the HPO plot is too busy due to many cases with variable phenotype please refer to the HPO table or the “hpo” column in the variant or lolliplot tsv tables

- one `HPO tsv` table: table with tabulation of all the HPO terms found in cases and their frequency

● one `variants tsv` table containing all Exomiser variants in that gene including cases, controls, excluded and variants not contributing to the gene-test significance. * please note that some variants might match the different mode of inheritance.

[1] caco_definition Case-control definition (case= case with the event tested contributing to the signal, excluded= patient with the event tested but excluded from analysis (caco==NA), control= control with the event tested contributing to the signal, not_contributing_case= case without the event tested not contributing to the signal, not_contributing_excluded= patient without the event tested not contributing to the signal (?), not_contributing_control= control without the event tested not contributing to the signal)

[2] caco Case-control (1=case, 0=control, NA=excluded)

- **Expected run time for demo on a "normal" desktop computer**

Instructions for use

Prepare your own versions of the input files (as below) and then run as described above for the example files.

- **exomiser_master_file_passvars.tsv** summarising the Exomiser variant output and proband data for each of the samples to be analysed in your cohort. This can be achieved by running:
    a. `perl produce_generic_exomiser_master_file_final.pl <samples file> <optional de novo file> <optional CCR file> <optional assembly: b37 or b38 (default)>`
        i. Samples file is the list of sampleIDs used to name Exomiser output files with the full path prefix if not in the current directory (one per line)
        ii. De novo file is a tab-sep file with sampleID, chr, pos, ref, alt for de novo called variants (one per line)
        iii. CCR file contains constrained coding regions in chr:start-end format (one per line)
2. **analysisLabelList.tsv** is the list of analyses to be run. It requires two columns: `analysis.label` and `analysis`
3. **analysis.label.tsv** are the case-control definitions. One per each `analysis.label` are required. It demands two columns: `sample.id`, `caco` and optionally `caco.denovo` (restricted to family.size >=3). The columns `caco` and `caco.denovo` defines the classification of cases, controls, and probands excluded from either. These columns assume values of 1, 0, or NA accordingly.