Readme

Zichen Zhang

July 22, 2022

1 TestAssociation.R

1.1 Typical run

TestAssociation.R is the *R* script that *SUMMIT* used to test the association between phenotypes and gene expression levels. Following is a typical run of *TestAssociation.R*. Usually, when testing association on one gene, *TestAssociation.R* takes less than 30 seconds.

Command Line

```
$ cd SUMMIT-test
$ Rscript TestAssociation.R \
$ --path.ref data/1000G.EUR.ALLSNP.QC.CHR \
$ --trait COVID-B2-V5 \
$ --path.out COVID-B2-V5 \
$ --parallel 3 \
```

1.2 Flags

- *path.ref*

  Name of the folder that contains the reference panel data.

- *trait*

  Name of the summary statistics file of the trait (phenotype) of interest.

- *path.out*

  Name of the folder to store all the output files.

- *parallel*

  Total number of parallel instances.

1.3 Data

TestAssociation.R requires the summary statistics of the trait of interest as its only exterior input. To run *TestAssociation.R* smoothly, your summary statistics **must** contain the following columns: *CHR, POS, A1, A2, SNP, Z*. In addition, the processed summary statistics **must** be split into smaller files per chromosome and each smaller files **must** be indexed by its chromosome (for example, *COVID-B2-V5-22.sumstats*). We recommend you try our interactive processing tool, *APSS*, as it is tailor-made for *SUMMIT*. More information on *APSS* can be found in the appendix section.

Comment: Of note, the example run that we showed you is a minimal reproducible example. For a complete reproduction of our results, please reach out to us or download the complete model from our Zenodo repository (`https://doi.org/10.5281/zenodo.6869129`).

1.4 Expected output

In our example, *COVID-B2-V5* in the *expected-output* folder is the expected output of *TestAssociation.R*. Following is a detailed breakdown of the output file.

Column name	Description	Value
gene_symbol	Gene name	*OAS1*
gene_id	Ensembl ID	ENSG00000089127
chromosome	Chromosome	12
model_best	Best-performing model	MCP
r2_test	Prediction R^2 on testing data	0.05605420
p_ElNet	p-value of Elastic Net model	2.436361×10^{-6}
z_ElNet	z-score of Elastic Net model	-5.37698890
...	...	...
p_ACAT	The combined ACAT p-value	1.377564×10^{-7}
gene_pos	Physical position of the gene	113357286
runtime	Runtime (in seconds)	6.227

Comment: *COVID-B2-V5* is a plain-text file with no file extension. You can also download complete real-data studies results in our manuscript from `https://chongwulab.shinyapps.io/SUMMIT-app/`.

2 Simulation.R

2.1 Typical run

Simulation.R is the *R* script that we used to generate **all** the simulation results in our manuscript. Following is a typical run of *Simulation.R*. Depending on your computing environment, running *Simulation.R* for one time would take 10 to 30 minutes.

Command Line

```
$ cd SUMMIT-test
$ Rscript Simulation.R \
$ --h2_e 0.01 \
$ --h2_p 0.2 \
$ --p_causal 0.01 \
$ --n 10000 \
$ --sumstats TRUE \
$ --gene_ENSG ENSG00000258289 \
$ --UKB TRUE \
$ --folder_output SIM-test \
$ --t1e FALSE \
$ --seed 1 \
```

2.2 Flags

- *h2_e*

  Heritability of simulated expression levels.

- *h2_p*

  Heritability of simulated phenotypes.

- *p_causal*

  Percentage of causal SNPs within a gene.

- *n*

  Sample size of the reference panel.

- *sumstats*

  Logical. If *sumstats* is TRUE, then *Simulation.R* would estimate X^TY using mock summary-level data generated by marginal regression and estimate X^TX using individual-level genotype data.

- *gene_ENSG*

  Ensembl ID of the gene being used in the simulation.

- *UKB*

  Logical. If you do not have access to UK BioBank individual-level genotype data, please set *UKB* to FALSE and *Simulation.R* will base all the simulation procedures on mock genotype matrices.

- *folder_output*

  Folder to store the results.

- *t1e*

  Logical. If *t1e* is set to TRUE, *Simulation.R* will explore the cases where the true association between phenotypes and expression level is set to 0. Most of the time, we recommend you set this to FALSE.

- *seed*

  The random seed being used.

> **Comment:** Specifically, in our practice, we paralleled our simulations into $1,000$ sub-jobs. For each sub-job, to avoid collision, it was assigned a unique random seed,
>
> $$R = (seed - 1) \times 1000 + i,$$
>
> where i is the sub-job's index, *seed* is the number you assigned to the *seed* flag.
>
> Usually, i would need to be explicitly passed on to *R* from the global environment (we worked on *SLURM*-managed computation platform). Depending on your computing environment, you may need to manually modify line 55 in *Simulation.R*.

2.3 Expected output

In our example, *101.RData-106.RData* in the *expected-output* folder is the expected output of *Simulation.R*. Following is a detailed breakdown of the output *.RData* file.

- *weight*

  weight contains all the inferred imputation models. It is a p-by-8 matrix. Its column names are the corresponding method being used. The column names are MCP, LASSO, ElNet, MNet, SCAD, Lassosum, TWAS-fusion, and PrediXcan.

- *r2*

  r2 is a vector containing all the testing R^2s and is ordered in accordance with *weight*.

- *out* is a matrix containing all the p-values of association tests and is ordered in accordance with *weight*.

2.4 Additional Comments

It can be very time-consuming to reproduce the complete simulation results in our manuscript with limited computation resources. Thus, **it is highly recommended that you run *Simulation.R* parallelly**.

Simulation.R only covered the first portion of our simulation studies (i.e., identifying expression imputation models). The sequential TWAS power studies can be properly handled by *TestAssociation.R*.

Of note, the genotype data we provided in our simulation example were simulated from a multinomial distribution to avoid violation of data confidentiality, whereas in the actual simulation study we conducted, **we leveraged UK Biobank data to construct the genotype matrix**.

In case you run into errors such as "plink/gcta_nr_robust could not be executed", please try the following commands.

Command Line

```
$ cd SUMMIT-test/software/
$ chmod +x plink
$ chmod +x gcta_nr_robust
```

3 Mainbody.R

3.1 Typical run

Mainbody.R is the *R* script that *SUMMIT* used to train imputation models. Following is a typical run of *Mainbody.R*. It usually takes less than 3 minutes to train imputation model for one gene.

Command Line

```
$ cd SUMMIT-test
$ Rscript Mainbody.R --name_batch TestRun1 --method SCAD
```

3.2 Flags

- *name_batch*

  Name of the output folder.

- *method*

  The type of penalized regression you wish to use. You can choose from LASSO, Lassosum, ElNet, SCAD, MCP, and MNet.

> **Comment:** Lassosum was added to *Mainbody.R* in our recent revision as one of the benchmark methods.

3.3 Data

Mainbody.R requires a very specific set of data to work properly. Unfortunately, we can not share all the data with you due to privacy and confidentiality concern. Following is a list of the data that we used. Items that are marked with an asterisk (*) are the ones that we can not provide and were replaced with simulated data.

- *gencode.v26.hg19.genes.rds*

  A table that we referenced to find corresponding Ensembl ID for each gene.

- *summary-statistics/eQTLGen*

  A folder that contains the summary statistics from eQTLGen (Tissue: whole blood, with standard quality control steps, subsetted by HapMap3). Due to the size of the original file (≥ 10Gb), we have split the raw data into smaller files (one file for each gene).

- *1000G.EUR.ALLSNP.QC.CHR*

  Reference data from The 1000 Genomes Project.

- *chr.OMNI.interpolated_genetic_map*

  A table that contains the genetic distance information. We utilized this information to achieve a better estimation of the LD matrix. You can manually turn this off by changing the *do.adjust* object to FALSE.

- *genotype.8.RData**

  A randomly-generated genotype matrix of all the GTex-8 subjects.

- *response.8.RData*

  A *R* object. *response.8.RData* is a matrix containing expression levels for all the GTEx-8 subjects.

3.4 Expected output

In our example, *Whole_Blood.ENSG00000089127.wgt.RData* in the *expected output* folder is the expected output of *Mainbody.R*. Worthy of noting, our imputation models are formatted in accordance with *TWAS-fusion* and hence our imputation model can be plugged into *TWAS-fusion*'s pipeline. Following is a detailed breakdown of the output *.RData* file.

- *cv.performance*

  cv.performance contains the adjusted prediction R^2 on testing data.

> **Comment:** Although the *R* object is named “cv.performance", our pipeline did not involve cross validation (named in accordance with *TWAS-fusion*).

- *snps*

  snps contains ancillary information (e.g. physical position, significance level in eQTLGen) about the gene and is ordered the same way as *wgt.matrix*.

- *wgt.matrix*

  wgt.matrix is a p-by-1 weight matrix, where p is the number of predictors(SNPs) in the gene.

> **Comment:** Since in our example run, we only used one type of penalized regression, hence *wgt.matrix* has only one column. In the actual model files we generated, *wgt.matrix* is a p-by-5 matrix.

Replication

We have made our best effort to ensure that all of our results are reasonably reproducible. In the *replication* folder, we have included a pdf file with extensive guidance (options and seeds) on how to fully replicate our results. In addition, the corresponding plotting codes were also included to help users reproduce the figures and tables exactly.

APSS

APSS is an interactive *R* function that can easily process GWAS summary statistics and shape GWAS summary statistics into any desired format. Listed below are *APSS*'s three principal input arguments.

- *directory.working*

  The working directory.

- *filename*

  The name of the summary statistics file to be processed.

- *BIG*

  BIG is the number of Gbs and default is 2. For example, if *BIG* is set as 5, then for any summary statistics file larger than 5Gb , *APSS* will do an exploratory read first. By doing so, *APSS* could significantly shorten the runtime and efficiently handle summary statistics files larger than 10Gb.

Dependencies, OS information, and versions

```
> print(version)
platform       x86_64-redhat-linux-gnu
arch           x86_64
os             linux-gnu
system         x86_64, linux-gnu
status
major          4
minor          1.0
year           2021
month          05
day            18
svn rev        80317
language       R
version.string R version 4.1.0 (2021-05-18)
nickname       Camp Pontanezen
> print(sessionInfo())
R version 4.1.0 (2021-05-18)
Platform: x86_64-redhat-linux-gnu (64-bit)
Running under: CentOS Linux 8

Matrix products: default
BLAS:   /R/R-4.1.0/lib64/R/lib/libRblas.so
LAPACK: /R/R-4.1.0/lib64/R/lib/libRlapack.so

attached base packages:
[1] stats     graphics  grDevices utils     datasets  methods   base

other attached packages:
[1] optparse_1.6.6    Rcpp_1.0.7        dplyr_1.0.7       ddpcr_1.15
[5] data.table_1.14.0 BEDMatrix_2.0.3

loaded via a namespace (and not attached):
 [1] fansi_0.5.0      assertthat_0.2.1 utf8_1.2.2       crayon_1.4.1
 [5] R6_2.5.1         DBI_1.1.1        lifecycle_1.0.0  magrittr_2.0.1
 [9] pillar_1.6.2     rlang_0.4.11     getopt_1.20.3    vctrs_0.3.8
[13] generics_0.1.0   ellipsis_0.3.2   crochet_2.3.0    glue_1.4.2
[17] purrr_0.3.4      compiler_4.1.0   pkgconfig_2.0.3  tidyselect_1.1.1
[21] tibble_3.1.4
```