traseR: TRait-Associated SNP EnRichment analyses

Li Chen, Zhaohui S.Qin

Department of Biostatistics and Bioinformatics
Emory University
Atlanta, GA 303022

li.chen@emory.edu,zhaohui.qin@emory.edu

October 14, 2015

Contents

Abstract

This vignette introduces the use of traseR (**TR**ait-**A**ssociated SNP**E**n**R**ichment analyses, which is designed to provide quantitative assessment whether a selected genomic interval(s) is likely to be functionally connected with certain traits or diseases. traseR consists of several modules, all written in R, to perform hypothesis testing, exploration and visualization of trait-associated SNPs(taSNPs). It also assembles the up-to-date taSNPs from dbGaP and NHGRI, SNPs from 1000 Genome Project CEU population with linkage disequilibrium greater than 0.8 within 100 kb of taSNPs, and all SNPs of CEU population from HapMap project into its built-in database, which could be directly loaded when performing analyses.

1 Introduction

Genome-wide association study (GWAS) have successful identified many sequence variants that are significantly associated with common diseases and traits. Tens of thousands of such trait-associated SNPs have already been cataloged which we believe is a great resource for genomic research. However, no tools existing utilizes those resources in a comprehensive and convenient way. In this study, we show the collection of taSNPs can be exploited to indicate whether a selected genomic interval(s) is likely to be functionally connected with certain traits or diseases. A R Bioconductor package named `traseR` has been developed to carry out such analyses.

2 Data collection

One great feature of `traseR` is the built-in database that collects various public SNP resources. Common public SNP databases include Association Result Browser, HapMap and 1000 Genome. We briefly introduce the procedures to process those public available SNP resources

2.1 Obtain taSNPs

Association Results Browser (http://www.ncbi.nlm.nih.gov/projects/gapplusprev/sgap_plus.htm) combines identified taSNPs from dbGaP and NHGRI, which together provide 44,078 SNP-trait associations, 30,553 unique taSNPs and 573 unique traits. This resource has been built into GRanges object `taSNPDB` and could be loaded into R console by typing `data(taSNPDB)`.

`traseR` need to specify the collection of trait-associated SNPs in particular format before we carry out enrichment analyses. The format starts with the columns,

1. Trait: Description of disease/trait examined in the study
2. SNP_ID: SNP rs number
3. p.value: GWAS reported p-values
4. seqnames: Chromosome number associated with rs number
5. ranges: Chromosomal position, in base pairs, associated with rs number
6. Context: SNP functional class
7. GENE_NAME: Genes reported to be associated with SNPs
8. GENE_START: Chromosome start position of genes
9. GENE_END: Chromosome end position of genes
10. GENE_STRAND: Chromosome strand associated with SNPs

Currently, the `traseR` package automatically synchronize trait-associated SNPs from Association Results Browser, which collects up-to-date GWAS results from dbGaP NHGRI GWAS catalog.

2.2 Obtain linkage disequilibrium taSNPs from 1000 Genome

We first download all vcf files from (ftp://share.sph.umich.edu/1000genomes/fullProject/2012.03.14/) that contains all genetic variants information. Then, two steps are used to find linkage disequilibrium SNPs >0.8 and located within 100kb of taSNPs. Firstly, we use `vcftools` to convert the vcf file format to `PLINK` format. Then we use `PLINK` to call the LD taSNPs by specifying options that limit the linkage disequilibrium SNPs >0.8 (–ld-window-r2 0.8) and within 100kb of taSNP (–ld-window-kb 100). The detailed commands are listed below,

```
vcftools -vcf vcf.file -out plink.file -plink plink -file plink.file -r2 -inter-chr -ld-snp-list
snps.txt -ld-window-r2 0.8 -ld-window-kb 100 -out output.file -noweb
```

Finally, we have 90700 SNP-trait associations and 78247 unique linkage disequilibrium trait-associated SNP. We also build linkage disequilibrium taSNP into another GRanges object `taSNPLDDB`, which could be loaded into R console by typing `data(taSNPLDDB)`.
The format of `taSNPLDDB` is,

1. seqnames: Chromosome number associated with rs number
2. SNP_ID: SNP rs number
3. ranges: Chromosomal position, in base pairs, associated with rs number
4. Trait: Description of disease/trait examined in the study

2.3 Obtain SNPs from HapMap

We download all genotype files in CEU population from the site, (http://hapmap.ncbi.nlm.nih.gov/downloads/genotypes/latest/forward/non-redundant/) The site contains merged phase I+II and III HapMap genotype files. We treat all alleles in the genotype files as SNPs for CEU population, which are expressed in the forward strand relative to the reference genome sequence. There are totally 4,029,840 variants excluding variants on Y chromosome, which could serve as background in hypothesis testing. Since all genotype files are in text format, we simply include all SNPs in auto chromosome and chromosome X into package built-in GRanges subject `CEU`.
The format of `CEU` is,

1. seqnames: Chromosome number associated with rs number
2. SNP_ID: SNP rs number
3. ranges: Chromosomal position, in base pairs, associated with rs number

3 Using `traseR`

To assess the enrichment level of trait-associated SNPs in given genomic interval(s) using `traseR`, one needs to follow the simple steps below.

1. Prepare the genomic intervals in data frame format with column names `chr`,`start`,`end`

2. Query a given a set of genomic interval(s) against all the taSNPs in the collection, perform statistical analyses
3. Explore genes/SNPs of particular interest

3.1 Background selection

3.1.1 Whole genome

The assumption is each base could be possibly be the taSNP. Based on the assumption. With the number of taSNPs inside and outside the genomic interval(s), the number of bases inside and outside of the genomic interval(s), we could classify all bases based on a base is taSNP or not and a base is in genomic intervals or not.

3.1.2 All SNPs

The assumption is each SNP could be possibly be the taSNP. Based on the assumption. With the number of taSNPs inside and outside the genomic interval(s), the non-taSNPs inside and outside of the genomic interval(s), we could classify all SNPs based on a SNP is taSNP or not and a SNP is in genomic intervals or not.

3.2 Hypothesis testing

traseR provides differential hypothesis testing methods in core function traseR, together with other functions for exploring and visualizing the results. The genomic interval(s) could be a data frame with three columns as chr(chromosome), start(genomic start position) and end(genomic end position). traseR also offers either using the whole genome or all SNPs as the background for hypothesis testing. If using whole genome as background, the command line is:

```
> x=traseR(snpdb=taSNPDB,region=Tcell)
> print(x)
```

If using all SNPs as background, the command line is:

```
> x=traseR(snpdb=taSNPLDDB,region=Tcell)
```

For the above commands, region is the data frame; snpdb is taSNPs; snpdb.bg is a background non-taSNPs; If rankby is set as "pvalue", all traits will be sorted by p-value in increasing order; ifrankby is set as "odds.ratio", all traits will be sorted by odds ratio in decreasing order. There are four options for test.method including "binomial", "chisq", "fisher", and "nonparametric"to perform binomial test, χ^2 test, Fisher's exact test and nonparametric respectively. If alternative is set to "greater", trader will perform hypothesis testing on whether genomic intervals are enriched of taSNPs than the background; If alternative is set to "less",traseR will perform hypothesis testing on whether genomic intervals are depleted of taSNPs than the background.

3.2.1 χ^2 and Fisher's exact test

Based on which background we choose, we could construct the 2 by 2 contingency table. then, we could perform χ^2 test on the table to assess the goodness of fit of the distribution of taSNPs inside and outside of genomic intervals(s).

We could also assume taSNPs in genomic intervals follows hypergeometric distribution and could calculate the p-value by using Fisher's exact test.

3.2.2 Binomial test

The assumption is the probability of observing a single base/SNP being a taSNP is the same inside and outside of genomic intervals. The probability of observing a single base/SNPs being a taSNP in genomic intervals could be estimated by using total number of taSNPs divided by the genome size/All SNPs. Then corresponding p-value could be calculated by using binomial test.

3.2.3 Nonparametric Test

Instead of enforcing any assumption, the control genomic interval(s) are generated by shuffling the genomic intervals randomly N times and overlap with taSNPs in each time. Then we could calculate the empirical p-value directly by counting how many taSNP hits larger/smaller than the observed taSNP hits.

3.3 Example

To further illustrate the usage of `traseR` R package, we download H3K4me1 peak regions in peripheral blood T cell from Roadmap Epigenomics. Those peak regions are deemed the genomic intervals. Since the degree of enrichment level is measured by p-value, we could rank traits based on p-value in an increasing order. We choose binomial test are the default option for `test.method`, use whole genome as background and over-enrichment as hypothesis testing direction.

```
> library(traseR)
> data(taSNPDB)
> data(Tcell)
> x=traseR(taSNPDB,Tcell)
> print(x)
```

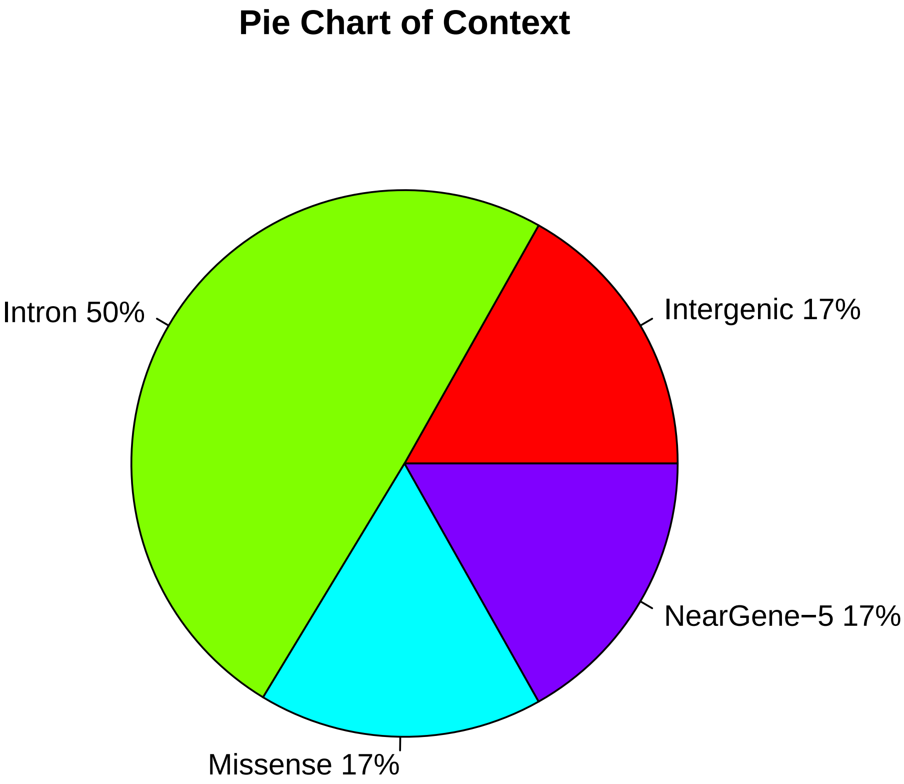
```
  Trait      p.value odds.ratio taSNP.hits taSNP.num
1   All 2.771864e-48   1.460704       1846     30553
                            Trait      p.value      q.value odds.ratio taSNP.hits
67                Behcet Syndrome 4.400406e-23 2.521433e-20   6.306579         59
172     Diabetes Mellitus, Type 1 1.704981e-11 4.884769e-09   5.045263         33
340 Lupus Erythematosus, Systemic 6.159346e-09 1.176435e-06   3.902195         32
49          Arthritis, Rheumatoid 1.442123e-07 2.065841e-05   5.126637         20
379            Multiple Sclerosis 1.644125e-05 1.884167e-03   2.905210         26
62           Autoimmune Diseases 5.201529e-05 4.967461e-03  15.892575          6
    taSNP.num
67        274
172       185
340       223
```

```
49          112
379         236
62           15
```

3.4 Exploratory and visualization functions

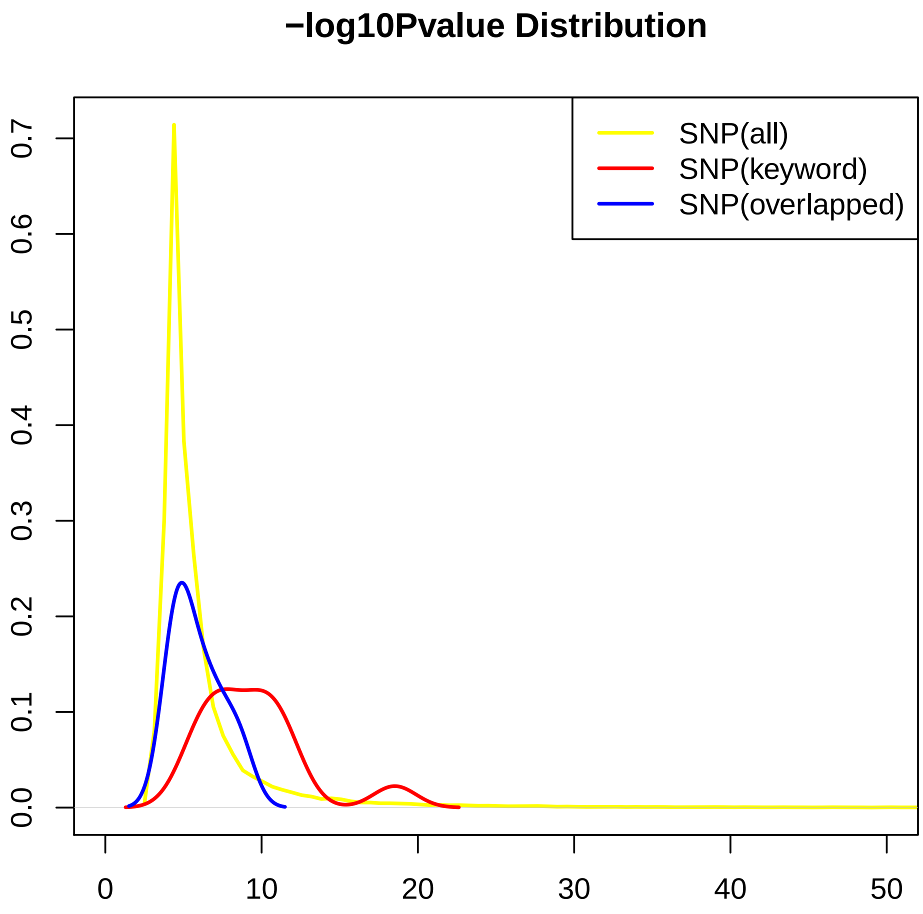
Plot the distribution of SNP functional class

```
> plotContext(snpdb=taSNPDB,region=Tcell,keyword="Autoimmune")
```



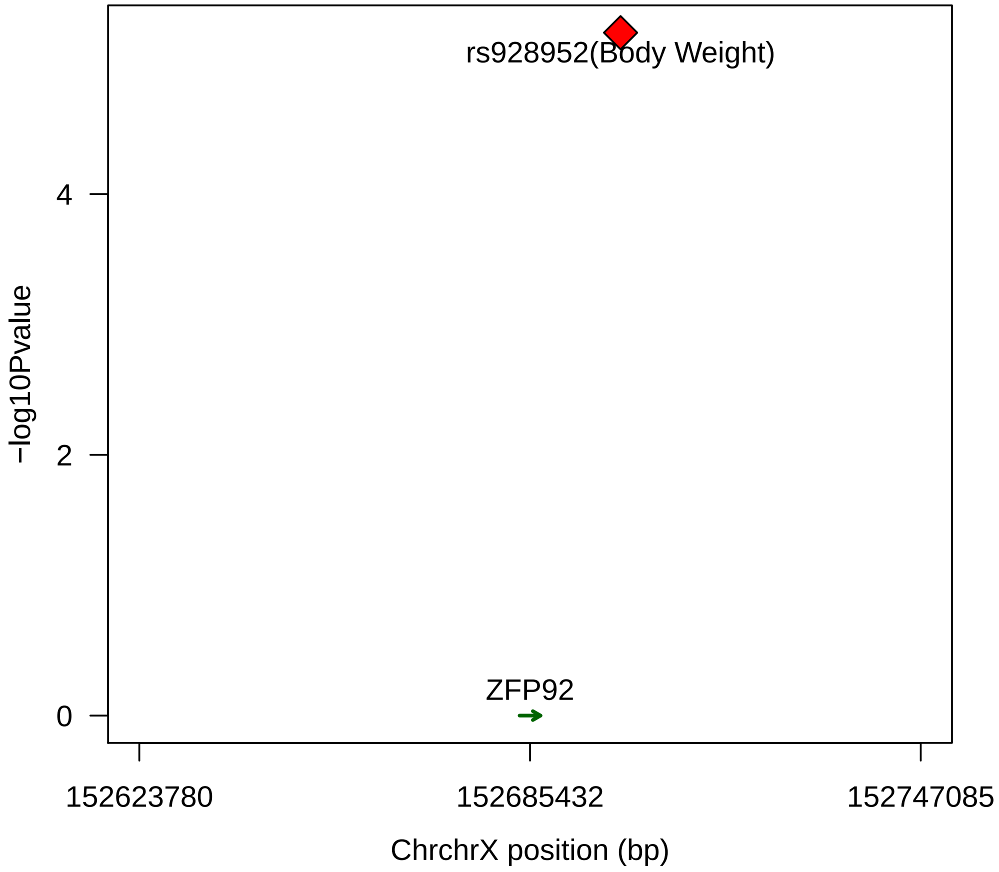
Plot the distribution of p-value of trait-associated SNPs

```
> plotPvalue(snpdb=taSNPDB,region=Tcell,keyword="autoimmune",plot.type="densityplot")
```



Plot SNPs or genes given genomic interval

```
> plotInterval(snpdb=taSNPDB,data.frame(chr="chrX",start=152633780,end=152737085))
```



Query trait-associated SNPs by key word,

```
> x=queryKeyword(snpdb=taSNPDB,region=Tcell,keyword="autoimmune",returnby="SNP")
> head(x)
```

```
          SNP_ID   Chr  Position Trait.num          Trait.name
42788 rs11203203 chr21  43836186         1 Autoimmune Diseases
4923   rs1876518  chr2  65608909         1 Autoimmune Diseases
26539  rs1953126  chr9 123640500         1 Autoimmune Diseases
42919  rs2298428 chr22  21982892         1 Autoimmune Diseases
4251   rs7579944  chr2  30445026         1 Autoimmune Diseases
2186    rs864537  chr1 167411384         1 Autoimmune Diseases
```

Query trait-associated SNPs by gene name,

```
> x=queryGene(snpdb=taSNPDB,genes=c("AGRN","UBE2J2","SSU72"))
> x
```

```
GRanges object with 3 ranges and 5 metadata columns:
      seqnames             ranges strand | GENE_NAME Trait.num          Trait.name
         <Rle>          <IRanges>  <Rle> |  <factor> <integer>            <factor>
  [1]     chr1 [ 955502,  991491]      + |      AGRN         1     Body Mass Index
  [2]     chr1 [1477052, 1510261]      - |     SSU72         1             Glucose
  [3]     chr1 [1189291, 1209233]      - |    UBE2J2         1 Waist Circumference
      taSNP.num taSNP.name
      <integer>   <factor>
  [1]         1  rs3934834
  [2]         1  rs3766178
  [3]         1 rs11804831
  -------
  seqinfo: 23 sequences from an unspecified genome; no seqlengths
```

Query trait-associated SNPs by SNP name,

```
> x=querySNP(snpdb=taSNPDB,snpid=c("rs3766178","rs880051"))
> x
```

```
GRanges object with 2 ranges and 8 metadata columns:
     seqnames             ranges strand |       Trait      SNP_ID   p.value     Context
        <Rle>          <IRanges>  <Rle> | <character> <character> <numeric> <character>
   9     chr1 [1478180, 1478180]      * |     Glucose   rs3766178  3.26e-05      Intron
  10     chr1 [1493727, 1493727]      * |     Glucose    rs880051  1.89e-05      Intron
       GENE_NAME GENE_START  GENE_END GENE_STRAND
     <character>  <integer> <integer> <character>
   9       SSU72    1477052   1510261           -
  10       SSU72    1477052   1510261           -
  -------
  seqinfo: 23 sequences from hg19 genome; no seqlengths
```

4 Conclusion

traseR provides methods to assess the enrichment level of taSNPs in a given sets of genomic intervals. Moreover, it provides other functionalities to explore and visualize the results.

5 Session Info

```
> sessionInfo()
```

```
R version 3.2.2 (2015-08-14)
Platform: x86_64-pc-linux-gnu (64-bit)
```

```
Running under: Ubuntu 14.04.3 LTS

locale:
 [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C               LC_TIME=en_US.UTF-8
 [4] LC_COLLATE=C               LC_MONETARY=en_US.UTF-8    LC_MESSAGES=en_US.UTF-8
 [7] LC_PAPER=en_US.UTF-8       LC_NAME=C                  LC_ADDRESS=C
[10] LC_TELEPHONE=C             LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C

attached base packages:
[1] stats4    parallel  stats     graphics  grDevices utils     datasets  methods
[9] base

other attached packages:
 [1] traseR_1.0.0                      BSgenome.Hsapiens.UCSC.hg19_1.4.0
 [3] BSgenome_1.38.0                   rtracklayer_1.30.0
 [5] Biostrings_2.38.0                 XVector_0.10.0
 [7] GenomicRanges_1.22.0              GenomeInfoDb_1.6.0
 [9] IRanges_2.4.0                     S4Vectors_0.8.0
[11] BiocGenerics_0.16.0

loaded via a namespace (and not attached):
 [1] XML_3.98-1.3               Rsamtools_1.22.0           GenomicAlignments_1.6.0
 [4] bitops_1.0-6               futile.options_1.0.0       zlibbioc_1.16.0
 [7] futile.logger_1.4.1        BiocStyle_1.8.0            lambda.r_1.1.7
[10] BiocParallel_1.4.0         tools_3.2.2                Biobase_2.30.0
[13] RCurl_1.95-4.7             SummarizedExperiment_1.0.0
```

References

[1] Welter D, MacArthur J, Morales J, Burdett T, Hall P, Junkins H, Klemm A, Flicek P, Manolio T, Hindorff L et al (2010). The NHGRI GWAS Catalog, a curated resource of SNP-trait associations. *Nucleic Acid Research*, **42**, D1001-1006.

[2] Roadmap Epigenomics C, Kundaje A, Meuleman W, Ernst J, Bilenky M, Yen A, Heravi-Moussavi A, Kheradpour P, Zhang Z, Wang J et al. (2015). Integrative analysis of 111 reference human epigenomes *Nature*, **7539**, 317-330

[3] http://www.ncbi.nlm.nih.gov/projects/gapplusprev/sgap_plus.htm *Association Results Browser*

[4] http://hapmap.ncbi.nlm.nih.gov/downloads/genotypes/latest/forward/non-redundant/ *HapMap*

[5] ftp://share.sph.umich.edu/1000genomes/fullProject/2012.03.14/ *1000Genome EUR*