1 Data

Abstract

In this paper we apply our methods to the Cancer Genetic Markers of Susceptibility (CGEMS) Genome-Wide Association Study (GWAS). The CGEMS dataset includes 1145 breast cancer cases - all postmenopausal women of European ancestry with invasive breast cancer - and 1142 controls from the Nurses' Health Study. The subjects have been genotyped using the Illumina HumanHap550 array, which provides the number of minor alleles (0, 1, or 2) for each subject at approximately 550000 Single Nucleotide Polymorphisms (SNPs). Additional covariate information available for each subject includes their age and use of postmenopausal hormones.

Availability

Due to confidentiality concerns, access to the CGEMS dataset is restricted and only available through the National Institutes of Health (NIH) database of Genotypes and Phenotypes (dbGaP). NIH employees, extramural principal investigators, and grantees with an electronic Research Administration (eRA) Commons account may submit a data access request to download the full dataset. The application is free, and if it is granted researchers may download the full dataset used in this manuscript.

To ease the readability of our provided code, we attach a mock version of the CGEMS dataset which can be used in place of the real data when running our data analysis script. Running our code on the mock data will not produce the results in the manuscript, but if the mock version is replaced with the actual data, then running our code will indeed reproduce Table 2 in the main paper.

Description

The dbGaP Study Accession identifier is phs000147.v1.p1, and the web page for the study is `https://www.ncbi.nlm.nih.gov/projects/gap/cgi-bin/study.cgi?study_id=phs000147.v1.p1`. This page also includes the link to request access. Genotype data are provided in PLINK binary format (.bed/.bim/.fam extensions), and additional covariates are provided in a clearly-labeled text file. We mimic the structure of the real dataset in our mock version. Due to our data use agreement, we are unable to provide more detailed information about the dataset.

2 Code

Abstract

Our major software contribution is the development of an R package `GBJ` which contains all the functions necessary to run our proposed tests and is available on CRAN. To ensure the reproducibility of our work, we additionally attach all the code used to generate each of the figures and tables in our manuscript. Due to the extensive length of our simulations, it was necessary to run all software on a high performance computing cluster. Workflow

instructions, cluster submission scripts, and other necessary files are all also attached with the code.

Description

The three major innovations proposed in our manuscript are (1) the construction of the Generalized Berk-Jones (GBJ) statistic for set-based inference; (2) a finite sample p-value calculation for supremum-based global tests (such as GBJ, Higher Criticism, and Generalized Higher Criticism); (3) an omnibus statistic incorporating multiple different global tests. These three techniques, along with some complementary functions, have been compiled into the R package `GBJ`. This package is freely available on CRAN and is made available under the GPL-3 license. `GBJ` version 0.4.0 was used to produce all the tables and figures in our submitted manuscript.

If accepted, all the attached code for reproducing manuscript figures and tables will additionally be deposited to a public repository on the first author's GitHub page. The workflow instructions and code to control job submission on a cluster will also be posted in the same repository, under a project dedicated to this paper. All of this reproducibility code will similarly be available under the GPL-3 license.

Figures 1, 2, and 3 and Table 2 will be reproduced exactly using the attached code and workflows described below. Supplementary Figures 3, 7, and 8 and Supplementary Tables 3, 4, and 5 will similarly be reproduced exactly by following our instructions. Figure 4 and Table 1 rely on HAPGEN2 to generate genotypes, and HAPGEN2 sets a random seed according to the time of day, so it is not possible to control the exact sequences of simulations for these results. Supplementary Figures 1, 2, 4, 5, and 6 similarly rely on HAPGEN2. However, since we are performing such a large number of simulations, the use of a different seed in random number generation will not materially affect the results or conclusions we draw. We have confirmed this point by performing the simulations multiple times. Following the instructions for Figure 4, Table 1, and Supplementary Figures 1, 2, 4, 5, and 6 will produce results that vary very little from those in the main manuscript.

Hardware and Software Requirements

The following is a list of all hardware, software, and additional reference files that are necessary to construct Figures 1-4 and Tables 1-2 in the main manuscript. While reference files arguably belong in the Data section, we believe it is more straightforward to place them here since they are not at all the focus of the analysis; these files act more like extensions of the software. Note that we will often refer to a working directory, which should be one folder that contains all the provided data/code/job submission scripts/reference files needed to recreate a given figure or table.

- Linux operating system.
- R version 3.1 or greater.
- The R libraries `GBJ`, `Rcpp`, `mvtnorm`, `stats`, `SKAT`, `bindata`, `dplyr`, `magrittr`, and `data.table`.

- Access to a computing cluster. We provide job submission scripts for LSF (.bsub extension) and SLURM (.sbatch extension) environments. You may need to edit these scripts slightly to ensure that relevant parameters (number of cores, amount of time requested, etc.) are in line with your computing facility rules. In particular, you will likely need to edit the queue name (the string after the '-q' flag) to reflect the actual name of your partition.

- The Boost C++ libraries (needed to compile the next item). Download them at `http://www.boost.org/users/history/version_1_65_1.html` and unzip the files.

- A C++ binary for calculating supremum-based global test p-values. The source code (`ebb_crossprob_cor.cpp`) for this binary is attached with the code for Figure 1. Put the .cpp file as well as the associated Makefile in your working directory. Change the second line of the Makefile to point to the directory which holds the 'boost' folder that was unzipped from above. Then navigate to the working directory from the command line, type `make`, and press ENTER to build the binary. (Figure 1 only)

- The HAPGEN2 binary (download at `http://mathgen.stats.ox.ac.uk/genetics_software/hapgen/hapgen2.html`) in your working directory. (Figure 4, Table 1, and Supplementary Figures 1, 2, 4, 5, and 6)

- Reference data from the HapMap3 project (download at `https://mathgen.stats.ox.ac.uk/impute/data_download_hapmap3_r2.html`) in your working directory. Specifically, you will need the chromosome 5 files `hapmap3.r2.b36.chr5.legend`, `genetic_map_chr5_combined_b36.txt`, and `CEU.chr5.hap`. You will also need the corresponding chromosome 10 files, where the string `chr5` is replaced by `chr10`. (Figure 4, Table 1, and Supplementary Figure 1)

- The 'glist-hg18.txt' file (attached with code for Table 1) in your working directory. This table defines gene positions according to the hg18 build of the human genome. (Tables 1 and 2)

- The PLINK v1.90 binary (download at `https://www.cog-genomics.org/plink2`) in your working directory. (Table 2 only)

- Genotypes at ten top genes as well as principal components for the CEU population of the 1000 Genomes Project (both attached with the code for Supplementary Table 5) in your working directly. (Supplementary Table 5 only)

3 Instructions for Use

Reproducibility

All instructions below are also attached in separate text files corresponding to each figure and each table. All R scripts and job submission files mentioned below are similarly attached. It is expected that all the scripts, files, and other necessary items for each figure or table are located in the current working directory when following the instructions for that figure or table.

All Figures

- The first line of each R script sets either the working directory or the directory holding necessary R libraries that should be installed. You will need to change this line in each script to reflect your own unique file paths. No other changes to the R scripts are necessary.

Figure 1

1. From the command line of your cluster computing environment, type either `bsub <run_GOF_bounds.bsub` or `sbatch run_GOF_bounds.sbatch` and press ENTER.
2. Once all the jobs from step (1) have finished, run `plot_fig1.R` from within an R console.

Figures 2 and 3

1. From the command line, type either `bsub <run_fig2_part1.bsub` or `sbatch run_fig2_part1.sbatch` and press ENTER.
2. Repeat step (1) with the .bsub/.sbatch scripts named `run_fig2_part2` through `run_fig2_part8`.
3. Once all the jobs from steps (1)-(2) have finished, run `plot_fig2_part1.R` from within an R console.
4. Repeat step (3) with all the scripts named `plot_fig2_part2.R` through `plot_fig2_part4.R`.

Figure 4

1. From the command line, type either `bsub <run_power_sim_chr5.bsub` or `sbatch run_power_sim_chr5.sbatch` and press ENTER.
2. Once all the jobs from step (1) have finished, run `plot_fig4.R` from within an R console.

Table 1

1. From the command line, type either `bsub <run_tab1_part1.bsub` or `sbatch run_tab1_part1.sbatch` and press ENTER.
2. Then type either `bsub <run_tab1_part2.bsub` or `sbatch run_tab1_part2.sbatch` and press ENTER.
3. Run `gen_data_tab1_part3.R` from within an R console.
4. Once all the jobs from steps (1), (2), and (3) have finished, run `make_tab1.R` from within an R console. The results from Table 1 will be stored in the variables `results_tab1`, `results_tab2`, and `results_tab3`.

Table 2

We have attached a mock version of the CGEMS dataset. Running the provided code with the mock dataset will NOT reproduce the results of Table 2. In fact, running with the mock dataset will produce mostly NA in the results table, because the mock dataset only contains data at a very small number of markers. To see more numerical results, one can edit the 'glist-hg18.txt' file to include more fake genes in the region covered by our fake dataset (chromosome 10, between 121001183-121876520). However, running the code with the actual dataset (using the file naming conventions of our fake datasets) will reproduce Table 2.

1. From the command line, type either `bsub <run_tab2.bsub` or `sbatch run_tab2.sbatch` and press ENTER.
2. Once all the jobs from step (1) have finished, run `make_tab2.R` from within an R console.

Supplementary Figures 1 and 2

1. From the command line, type either `bsub <run_suppfig1_part1.bsub` or `sbatch run_suppfig1_part1.sbatch` and press ENTER.
2. Then type either `bsub <run_suppfig1_part2.bsub` or `sbatch run_suppfig1_part2.sbatch` and press ENTER.
3. Once all the jobs from steps (1) and (2) have finished, run `plot_suppfig1_and2.R` from within an R console.

Supplementary Figure 3

1. Make sure you have collected in your data directory all the files originally needed to make Figure 1.
2. Run `plot_suppfig3.R` from within an R console.

Supplementary Figures 4, 5, and 6

1. From the command line, type either `bsub <run_suppfig4.bsub` or `sbatch run_suppfig4.sbatch` and press ENTER.
2. Repeat step (1) with the `run_suppfig5` and `run_suppfig6` files.
3. Once all the jobs from steps (1) and (2) have finished, run `plot_suppfig456.R` from within an R console.

Supplementary Figures 7 and 8

1. Make sure you have collected in your data directory all the files originally needed to make Table 2.
2. Run `plot_suppfig78.R` from within an R console.

Supplementary Tables 3 and 4

1. From the command line of your cluster computing environment, type either `bsub <run_suppfig3_part1.bsub` or `sbatch run_suppfig3_part1.sbatch` and press ENTER.
2. Repeat step (1) with the .bsub/.sbatch scripts marked parts 2-4 and also those starting with `run_suppfig4`.
3. Once all the jobs from steps (1)-(2) have finished, run `make_supptab34.R` from within an R console.

Supplementary Table 5

Running the provided code with the mock dataset will NOT reproduce the results of Supplementary Tables 5. Running the provided code with the real CGEMS dataset will reproduce the results of Supplementary Tables 5.

1. Run `make_supptab5.R` from within an R console.

Replication

Instructions for replication are covered extensively in the documentation, README, and vignette of the `GBJ` package.