

# Author Contributions Checklist Form

This form documents the artifacts associated with the article (i.e., the data and code supporting the computational findings) and describes how to reproduce the findings.

## Part 1: Data

☐ This paper **does not** involve analysis of external data (i.e., no data are used or the only data are generated by the authors via simulation in their code).

☒ I certify that the author(s) of the manuscript have legitimate access to and permission to use the data used in this manuscript.

## Abstract

We used the imputed genotype data from UK Biobank (UKB) with Field ID of 22828 and resting-state functional fMRI (rfMRI) partial correlation matrices with Field ID of 25753 in our paper.

## Availability

☒ Data **are** publicly available

☐ Data **cannot be made** publicly available

If the data are publicly available, see the *Publicly available data* section. Otherwise, see the *Non-publicly available data* section, below.

## Publicly available data

☐ Data are available online at:

☐ Data are available as part of the paper's supplementary material.

☒ Data are publicly available by request, following the process described here:

All Data are publicly accessible from UK Biobank via their standard data access procedure at <https://www.ukbiobank.ac.uk/>. Researchers can apply for access to the UK Biobank data via the Access management System (AMS) at <http://www.ukbiobank.ac.uk/enable-your-research/apply-for-access>.

☐ Data are or will be made available through some other mechanism, described here:

## Non-publicly available data

Discussion of lack of publicly available data:

## Description

### File format(s)

- ☒ CSV or other plain text:
- ☒ Software-specific binary format (.Rda, Python pickle, etc.):
- ☐ Standardized binary format (e.g., netCDF, HDF5, etc.):
- ☐ Other (described here):

### Data dictionary

- ☐ Provided by the authors in the following file(s):
- ☐ Data file(s) is (are) self-describing (e.g., netCDF files)
- ☒ Available at the following URL:

<http://www.ukbiobank.ac.uk/>

### Additional information (optional)

## Part 2: Code

### Abstract

We have included programming codes to implement the methods and codes to run simulations, generate outcome values and analyze the data.

### Description

#### Code format(s)

☒ Script files

☒ R ☐ Python ☐ Matlab

☐ Other:

☐ Package

☐ R ☐ Python ☐ MATLAB toolbox

☐ Other:

☒ Reproducible report

☒ R Markdown ☐ Jupyter notebook

☐ Other:

☐ Shell script

☐ Other (described here):

### Supporting software requirements

#### Version of primary software used

R version 4.2.0

#### Libraries and dependencies used by the code

R.matlab (3.6.2), huge (1.3.5), Rcpp (1.0.8.3), Ball (1.3.13), snpStats (1.4 8.0), Matrix (1.5.4), survival (3.3-1).

## Supporting system/hardware requirements (optional)

Click or tap here to enter text.

### Parallelization used

- ☒ No parallel code used
- ☐ Multi-core parallelization on a single machine/node  
Number of cores used: 4
- ☐ Multi-machine/multi-node parallelization  
Number of nodes and cores used:

### License

- ☒ MIT License (default)
- ☐ BSD
- ☐ GPL v3.0
- ☐ Creative Commons
- ☐ Other (described here):

### Additional information (optional)

## Part 3: Reproducibility workflow

### Scope

The provided workflow reproduces:

- ☒ Any numbers provided in text in the paper
- ☒ The computational method(s) presented in the paper (i.e., code is provided that implements the method(s))
- ☒ All tables and figures in the paper
- ☐ Selected tables and figures in the paper, as explained and justified here:

### Workflow details

#### Location

The workflow is available:

- ☐ As part of the paper's supplementary material
- ☒ In this Git repository: <https://github.com/daiw3/BCRA.git>
- ☐ Other:

#### Format(s)

- ☐ Single master code file
- ☐ Wrapper (shell) script(s)
- ☒ Self-contained R Markdown file, Jupyter notebook, or other literate programming approach
- ☐ Text file (e.g., a readme-style file) that documents workflow
- ☐ Makefile
- ☐ Other (more detail in 'Instructions' below)

#### Instructions

##### **Instructions to Perform Simulations**

##### **Step 1: Generate simulated Dataset**

`./demo/simulation/0-single-snpset-generate-simu-data.R` is the code to generate simulated dataset using real genotype dataset from UKB for single SNP-set simulation. It has multiple parameter inputs, including nonlinear settings (`non_linear_setting`), error correlation structures (`graph`), proportion of true signals (`p_snp`), proportion of subsets used in subsample-BCRA (`sub_p`), distance measurements (`dist_name`), noise level (`noise_level`)

Given data privacy and capacity constraints, we did not put the original genotype data online. We generated a demo dataset `./data/simu_data_demo.RData` based on the this code for illustration purpose.

`./demo/simulation/0-multi-snpset-generate-simu-data.R` is a demonstration code to generate synthetic data using real genotype dataset from UKB for Multi SNP-set simulation.

Given data privacy and capacity constraints, we did not put the original genotype data online. We generated a demo dataset `./data/simu_data_multiset_demo.RData` based on the this code for illustration purpose.

### Step 2a: Run Single-SNP set simulation

`./demo/simulation/1-simu_single_snpSet_subsample_BCRA.R` is a demonstration code to repeat the single SNP-set simulation for subsample-BCRA using the demo simulated dataset. It will generate result for 1 iteration. We put an example output under `./results/single_SNPset/`

`./demo/simulation/1-simu_single_snpSet_BCRA_GWAS.R` is a demonstration code to repeat the single SNP-set simulation for BCRA and GWAS using the demo simulated dataset. It will generate result for 1 iteration. We put an example output under `./results/single_SNPset/`.

### Step 2b: Run Multi-SNP set simulation

`./demo/simulation/1-simu_multi_snpSet_subsample_BCRA.R` is a demonstration code to repeat the single SNP-set simulation for both BCRA and subsample-BCRA using the demo simulated dataset. We put an example output under `./results/multi_SNPset/`.

### Step 3: Summarize simulation results and replicate Tables and Figures in the manuscript

`./demo/simulation/2_reproduce_single_snpSet_simu.R` provides the code to calculate detection rate for single SNP-set simulation. We put one example output for the power simulation with  $\pi = 0.01$  (proportion of true signals) as in Fig.4 under the path `./results/Singleset_Figure4.RData`. This can reproduce the results of Fig.4 for  $\pi = 0.01$ .

`./demo/simulation/2_reproduce_multi_snpSet_simu.R` provides the code to calculate detection rate, SEN, SPE, PREC, NVP for Multi SNP-set simulation. We put one example output for the power simulation with the crossing case as in Table 2 under the path `./results/Multiset_Table1_3_S3_to_S9.RData`. This can reproduce the results of Table 1 to 3, S3 to S9.

`./demo/reproduce_figure_tables/*.R` contain the code to replicate all figures and tables in the simulation studies, including Fig.3-4, Fig.S1-S3, Fig.S5, Table S1 to S2.

### Real Data Application (UKB)

**Step 1: Calculate geodesic distance among functional connectivity matrices**

Calculate geodesic distance (other distance measures can be applied depending on your interest) of functional connectivity matrices across each pair of subjects.

**Step 2: Partition Genotype into SNP-set**

Partition Genotype into SNP-set: you can use gene/LD-block or physical locations (what we adopted) to do the partition. PLINK software can be used to fulfill the partition.

**Step 3: Genotype QC on each SNP-set**

Genotype QC: we exclude: 1) subjects with more than 10% missing genotypes; 2) variants with missing genotype rate larger than 10%; and 3) variants that failed the Hardy-Weinberg test at  $10e-6$  level.

**Step 4: Run subsample-BCRA for each SNP-set**

Run subsample-BCRA on a SNP-set using code located under `./demo/UKB/1-run-subsample-BCRA-SNPset.R`. Since we are unable to provide sensitive real data, we provided a demo dataset named as `*./data/chr5_8_demo.RData*` for illustration purpose to execute the code. The output of the code will give selected SNPs (`selected_snps_int`) and permutation p-value for this SNP-set `pval_results_perm`.

**Step 5: Summarize p-values associated with each SNP-set**

We put a demonstration

code `./demo/reproduce_figure_tables/FigureS1_S3_S5_Table_S1_S2.R` to plot p-values associated with each SNP-set (Figure S5) with the data

file `./demo/reproduce_figure_tables/pval_UKB.RData` gives the SNPs selected with permuted p-value for each SNP-set presented in the manuscript.

**Step 6: Visualize the results**

The connectivity plots in the manuscript (Figure 6 and Figure S6-S1) can be generated using `chordDiagram` function in R.

## Expected run-time

Approximate time needed to reproduce the analyses on a standard desktop machine:

- ☐ <1 minute
- ☐ 1-10 minutes
- ☐ 10-60 minutes
- ☐ 1-8 hours
- ☐ >8 hours
- ☒ Not feasible to run on a desktop machine, as described here:

**1. Single SNP-set Simulation**

Memory: 32 GB per CPU

CPUs: 4

Time: ~1 hour per iteration for one parameter set

Example: Generating results for ( $\pi = 0.01$ ) (proportion of true signals) as shown in Figure 4.

**2. Multi SNP-set Simulation**

Memory: 32 GB per CPU

CPU: 4

Time: ~40 hours per iteration for one parameter set for BCRA and ~4 hours per iteration under the same resource conditions for subsample-BCRA.

Example: Generating results for  $(\mathbf{B})$  in the "smile" case for BCRA.

### 3. UK Biobank Real Data Application

#### a. Distance Calculation

Memory: At least 100 GB per CPU

CPU: 4

Time: ~1 week

Note: Additional memory is required to write and store distance matrices.

#### b. BCRA/Subsample Analysis

Memory: 32 GB per CPU

CPU: 4

Time: BCRA: ~1 week per SNP-set with 5000 permutations. Subsample-BCRA: 2 days per SNP-set with 5000 permutations

### Additional documentation (optional)

--

### Notes (optional)

--