Journal of Statistical Software

MMMMMM YYYY, Volume VV, Issue II. *doi: 10.18637/jss.v000.i00*

MRBEE: R and Python Functionality for Robust Multivariable Mendelian Randomization using GWAS Summary Data

Noah Lorincz-Comi **Yihe Yang** **Xiaofeng Zhu**

Case Western Reserve University

Abstract

Multivariable Mendelian randomization (MVMR), which uses causal genetic variants as instrumental variables to remove the confounding effect, has become one of the most popular statistical methods to infer causal relationships between multiple exposures and outcomes. With the increasing number of participants in large genome-wide association studies, more and more causal genetic variants are being identified, making the weak instrument bias unignorable in current MVMR analyses. In this paper, we demonstrate how to remove the weak instrumental bias for common MVMR methods such as multivariable inverse-variance weighting, MVMR-Robust, and MVMR-Median. We then introduce our newly developed **R** package and **Python** module **MRBEE** which efficiently implements these bias-corrected MVMR methods in practice. The use of package **MRBEE** is illustrated and a comparison of the traditional MVMR methods is made through the analyses of multiple real datasets.

Keywords: Causal Inference, Genome-Wide Association Studies, Multivariable Mendedian Randomization, Weak Instrument Bias.

1. Introduction

Mendelian randomization (MR) is an instrumental variable (IV) method that leverages genetic variants as IVs to estimate causal effects of exposures on outcome phenotypes (Davey Smith and Ebrahim 2003). In comparison to traditional causal inference tools such as randomized controlled trials (RCTs), MR takes advantage of the natural randomness of genetic variation, which does not require the ethical and logistical challenges associated with RCTs. An alternative advantage of MR beyond traditional causal inference tools is that MR can be conducted

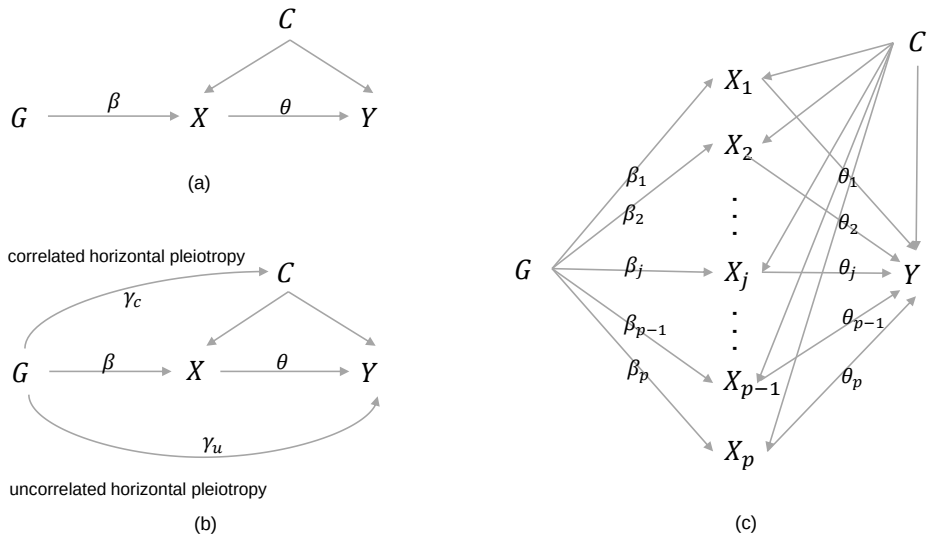

Figure 1: DAG of UVMR and MVMR. Panel (a): causal path diagram with valid genetic IVs. Panel (b): causal path diagram with UHP and CHP. Panel (c): causal path diagram for MVMR methods. G: genetic IVs; X: exposure; Y: outcome; C: confounders; β: association between G and X; θ: causal effect of X on Y; γ_c: direct correlation between G and C; γ_u: direct correlation between G and Y.

by only using genome-wide association studies (GWAS) summary data (Sanderson *et al.* 2022), which can be publicly accessed in open-source databases like dbGaP (Mailman *et al.* 2007) and GWAS catalog (MacArthur *et al.* 2017). In the post-GWAS era, MR has yielded insights into the etiology of a wide range of diseases from cardiovascular disease (Wang *et al.* 2022) and diabetes (Corbin *et al.* 2016) to schizophrenia (Vaucher *et al.* 2018) and cancer (Shen *et al.* 2021).

Inverse-variance weighting (IVW) is considered as the most basic method for performing MR with GWAS summary data (Burgess *et al.* 2013). The validity of IVW rests on three key conditions known as the valid IVs conditions: the genetic variants must (IV1) be strongly associated with the exposure, (IV2) have an association with the outcome that is solely through the exposure, and (IV3) be independent of confounders of the exposure-outcome relationship (Zhu 2020). Panel (a) in Figure 1 shows the directed acyclic graph (DAG) of valid IV conditions for MR with a single exposure. IVs that violate the (IV2) or (IV3) conditions are considered to have uncorrelated horizontal pleiotropy (UHP) or correlated horizontal pleiotropy (CHP), respectively (Morrison *et al.* 2020). Panel (b) in Figure 1 shows a DAG of UHP and CHP. UHP and CHP are common concerns in MR, and so many robust methods to eliminate their effects have been developed: MR-Egger (Bowden *et al.* 2015), MR-Median (Bowden *et al.* 2016), MR-Robust (Rees *et al.* 2019), MR-PRESSO (Verbanck *et al.* 2018), MRMix (Qi and Chatterjee 2019), CAUSE (Morrison *et al.* 2020), IMRP (Zhu *et al.* 2021), and MR-CUE (Cheng *et al.* 2022), and MR-APSS (Hu *et al.* 2022).

Multivariable MR (MVMR) includes multiple exposures in MR simultaneously and is used to estimate direct and unmediated causal effects on an outcome phenotype (Sanderson *et al.* 2019). A DAG displaying MVMR is presented in Panel (c) of Figure 1. MVMR can also be considered as a valid tool for addressing UHP and CHP bias in MR. That is, if the other

exposures through which an IV violates conditions (IV1) or (IV2) are included in MVMR, then (IV1) and (IV2) are not violated, making MVMR potentially more robust to UHP and CHP bias as more exposures are included. Until now, multivariable versions of IVW (Burgess and Thompson 2015), MR-Egger (Rees *et al.* 2017), MR-Median, and MR-Robust (Grant and Burgess 2021) have been developed.

However, MVMR is vulnerable to weak instrument bias which is often overlooked and challenging to address (Burgess *et al.* 2011). The traditional solution to weak instrument bias involves iteratively removing weak IVs until a small set of relatively strong IVs remains (Burgess *et al.* 2011; Sanderson *et al.* 2021). Recent studies have demonstrated that this approach may introduce selection bias and result in inefficient causal estimates (Sadreev *et al.* 2021). Two novel UVMR methods, MR robust adjusted profile score (MR-RAPS, Zhao *et al.* (2020)) and debiased IVW (DIVW, Ye *et al.* (2021)), were developed to address weak instrument bias without introducing extra selection bias. However, they suffer from limited practical applications because they can only accommodate a single exposure and require the exposure and outcome GWAS to be independent of each other. As larger GWAS are being performed, it is becoming increasingly difficult to find exposure and outcome GWAS samples that do not share any participants.

Our recently proposed method, MR using Bias-corrected Estimating Equation (MRBEE), represents a significant advancement in MVMR analysis, as it can effectively remove weak instrument bias while accommodating arbitrary sample overlap of multiple GWAS cohorts (Lorincz-Comi *et al.* 2023). We theoretically discovered that the commonly used multivariable IVW cannot generally provide asymptotically valid inferences. In contrast, MRBEE can provide unbiased causal effect estimates and asymptotically valid inferences in many real world conditions. However, MRBEE only addresses weak instrument bias in multivariable IVW. Whereas, geneticists and epidemiologists often prefer more robust methods due to the potential presence of UHP and CHP. It is unclear how weak instrument bias affects other robust MVMR methods, such as MVMR-Median and MVMR-Robust, but the strategy of removing the weak instrument bias by MRBEE can be applied to these MVMR methods.

Regarding the available statistical software, two R packages,**MendelianRandomization** (Yavorska and Burgess 2017) and **TwoSampleMR** (Hemani *et al.* 2020), offer functionality for performing MR with many basic methods. Newer UVMR methods intended to be more robust than these methods provide R code at their corresponding GitHub repositories; see, e.g., **MRMix** (Qing and Chatterjee 2019), **CAUSE** (Morrison 2019), and **MR.CUE** (Qing *et al.* 2022). While **TwoSampleMR** offers functionality for performing necessary quality control (QC) steps, it only includes common univariable approaches. On the other hand, **MendelianRandomization** offers both common univariable and multivaraible approaches such as IVW, MR-Median, and MR-Lasso, but it cannot be used to perform QC steps. Nevertheless, both **MendelianRandomization** and **TwoSampleMR** lack advanced MVMR methods that can address weak instrument bias and GWAS sample overlap, which is becoming a growing problem for MR due to the growing number of identified causal genetic variants for multiple exposures.

This paper presents two significant contributions. Firstly, we demonstrate how to remove the weak instrument bias in common MVMR approaches such as MVMR-Robust and MR-Median. These bias-corrected MVMR approaches offer valuable insights into statistical methodology because no prior studies have attempted to address this issue in MVMR approaches. Second, we introduce the **MRBEE** software that implements these bias-corrected MVMR approaches efficiently and in a user-friendly way. **MRBEE** requires only GWAS sum-

mary statistics and is used to perform allele harmonization, estimate causal effects between multiple exposures and a single outcome without bias, and perform genome-wide horizontal pleiotropy testing to find novel genetic variants (Lorincz-Comi *et al.* 2023; Zhu *et al.* 2022). The **MRBEE** software is available on Github as both an **R** package, **Python** module, and command line tool. There are currently no **Python** modules available to perform MR, making our module the first.

The paper is structured as follows. In Section 2, we present the novel statistical methodologies that address weak instrument bias for multivariable IVW, MVMR-Robust, MVMR-Median, and MVMR-Mixture. As these methods are newly developed, we conduct simulations to validate their effectiveness in removing weak instrument bias under different settings. In Section 3, we provide a comprehensive overview of the implementation of the MVMR methods described in Section 2 using the user-friendly **R** package, **MRBEE**. We also provide illustrations of the computational tools available in the package. Section 4 briefly demonstrates how to use the command-line version of the MRBEE **Python** software, and Section 5 concludes the paper with a summary of our findings and further discussions.

2. Multivariable Mendelian Randomization

2.1. MVMR model

Let $\boldsymbol{g}_i = (g_{i1}, \ldots, g_{im})^\top$ represent a vector containing the genotypes of m independent genetic variants, where each variant is standardized to have a mean of zero and a variance of one. In practice, the technique known as linkage disequilibrium (LD) clumping is commonly employed to acquire independent variants by selecting a single representative variant for each LD region (Purcell *et al.* 2007). Furthermore, let $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{ip})^\top$ represent a vector containing p exposures and y_i represent an outcome. Consider the following linear structural model:

$$\boldsymbol{x}_i = \mathbf{B}^\top \boldsymbol{g}_i + \boldsymbol{u}_i, \tag{1}$$

$$y_i = \boldsymbol{\theta}^\top \boldsymbol{x}_i + \boldsymbol{\gamma}^\top \boldsymbol{g}_i + v_i, \tag{2}$$

where $\mathbf{B} = (\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_m)^\top$ is an $(m \times p)$ matrix of genetic effects on exposures with $\boldsymbol{\beta}_j = (\beta_{j1}, \ldots, \beta_{jp})^\top$ being a vector of length p, $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_p)^\top$ is a vector of length p representing the causal effects of the p exposures on the outcome, $\boldsymbol{\gamma} = (\gamma_1, \ldots, \gamma_m)^\top$ is a vector of length m representing the violation of (IV2) and (IV3) conditions, and $\boldsymbol{u}_i$ and v_i are noise terms. Substituting for $\boldsymbol{x}_i$ in (2), we obtain the reduced-form model:

$$y_i = \boldsymbol{g}_i^\top \boldsymbol{\alpha} + \boldsymbol{u}_i^\top \boldsymbol{\theta} + v_i, \tag{3}$$

where $\boldsymbol{\alpha} = \mathbf{B}\boldsymbol{\theta} + \boldsymbol{\gamma}$. Since the random errors $\boldsymbol{u}_i$ and v_i are often correlated, traditional linear regression between $\boldsymbol{x}_i$ and y_i fails to consistently estimate the causal effect vector $\boldsymbol{\theta}$. Fortunately, the genetic variant vector $\boldsymbol{g}_i$ is considered to be independent of $\boldsymbol{u}_i$ and v_i, as individual genotypes are randomly inherited from parents and generally remain unaltered throughout one's lifetime (Davey Smith and Ebrahim 2003). As a result, $\boldsymbol{g}_i$ can be used as IVs to mitigate the confounding effects of $\boldsymbol{u}_i$ and v_i.

Unlike traditional statistical analyses, the individual-level data in GWAS are less available. As a result, most of the current MR analyses are performed with GWAS summary data

through the following linear regression:

$$\hat{\alpha}_j = \hat{\boldsymbol{\beta}}_j^\top \boldsymbol{\theta} + \gamma_j + \varepsilon_j, \tag{4}$$

where $\hat{\alpha}_j$ and $\hat{\boldsymbol{\beta}}_j$ are estimated from the outcome and exposure GWAS for the jth IV, γ_j is the horizontal pleiotropy, and ε_j represents the residual in this regression model. Let $w_{\alpha_j} = \hat{\alpha}_j - \alpha_j$ and $w_{\beta_{js}} = \hat{\beta}_{js} - \beta_{js}$ be the estimation errors of α_j and β_{js}, $j = 1, \ldots, p$. Lorincz-Comi *et al.* (2023) proved under moderate conditions that

$$\begin{pmatrix} \sqrt{n}_0 w_{\alpha_j} \\ \sqrt{n}_1 w_{\beta_{1j}} \\ \vdots \\ \sqrt{n}_p w_{\beta_{1p}} \end{pmatrix} \xrightarrow{D} \mathcal{N}\left(\begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \begin{pmatrix} \sigma_{yy} & \frac{n_{01}}{\sqrt{n_0 n_1}}\sigma_{yx_1} & \cdots & \frac{n_{01}}{\sqrt{n_0 n_p}}\sigma_{yx_p} \\ \frac{n_{01}}{\sqrt{n_0 n_1}}\sigma_{yx_1} & \sigma_{x_1x_1} & \cdots & \frac{n_{1p}}{\sqrt{n_1 n_p}}\sigma_{x_1x_p} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{n_{0p}}{\sqrt{n_0 n_p}}\sigma_{yx_p} & \frac{n_{1p}}{\sqrt{n_1 n_p}}\sigma_{x_1x_p} & \cdots & \sigma_{x_px_p} \end{pmatrix} \right),$$

where $\sigma_{x_j x_k} = \mathrm{cov}(x_{ij}, x_{ik})$, n_s is the sample size of the sth GWAS cohort (DO YOU MEAN STH EXPOSURE COHORT??), and n_{sk} is the overlapping sample size between the sth and the kth GWAS cohorts (x_{0j} represents y_i and the 0th cohort refers to the outcome GWAS cohort). In other words, the MVMR model (4) can be considred a linear regression model with normal measurement errors in the covariates and response (Yi 2017).

2.2. Weak instrument bias

In practice, we often standardize $\hat{\alpha}_j$ and $\hat{\beta}_{js}$ by $\hat{\alpha}_j/\mathrm{se}(w_{\alpha_j})$ and $\hat{\beta}_{js}/\mathrm{se}(w_{\beta_{js}})$ to effects of varying allele frequencies (Zhu *et al.* 2022). After standardizing in this way, the multivariable IVW estimates $\boldsymbol{\theta}$ by

$$\hat{\boldsymbol{\theta}}_{\mathtt{IVW}} = \arg\min_{\boldsymbol{\theta}} \left\{ \frac{1}{2} \sum_{j=1}^{m} (\hat{\alpha}_j - \hat{\boldsymbol{\beta}}_j^\top \boldsymbol{\theta})^2 \right\} = (\widehat{\mathbf{B}}^\top \widehat{\mathbf{B}})^{-1} \widehat{\mathbf{B}}^\top \hat{\boldsymbol{\alpha}}. \tag{5}$$

Lorincz-Comi *et al.* (2023) proved that even if the three valid IVs conditions are satisfied, $\hat{\boldsymbol{\theta}}_{\mathtt{IVW}}$ is still biased because the estimation errors w_{α_j} and $w_{\beta_{js}}$ inflate the weak instrument bias. To see this, consider the estimating equation and Hessian matrix of $\hat{\boldsymbol{\theta}}_{\mathtt{IVW}}$:

$$\boldsymbol{S}_{\mathtt{IVW}}(\boldsymbol{\theta}) = \sum_{j=1}^{m} (\hat{\boldsymbol{\beta}}_j^\top \boldsymbol{\theta} - \hat{\alpha}_j) \hat{\boldsymbol{\beta}}_j = \widehat{\mathbf{B}}^\top (\widehat{\mathbf{B}} \boldsymbol{\theta} - \hat{\boldsymbol{\alpha}}), \tag{6}$$

$$\mathbf{H}_{\mathtt{IVW}} = \sum_{j=1}^{m} \hat{\boldsymbol{\beta}}_j \hat{\boldsymbol{\beta}}_j^\top = \widehat{\mathbf{B}}^\top \widehat{\mathbf{B}}. \tag{7}$$

The multivariable IVW estimate is found by solving $\boldsymbol{S}_{\mathtt{IVW}}(\boldsymbol{\theta}) = \mathbf{0}$ for $\boldsymbol{\theta}$ and $\mathbf{H}_{\mathtt{IVW}} = \partial S_{\mathtt{IVW}}(\boldsymbol{\theta})/\partial \boldsymbol{\theta}$. Since $\hat{\boldsymbol{\theta}}_{\mathtt{IVW}} - \boldsymbol{\theta} = -\mathbf{H}_{\mathtt{IVW}}^{-1} \boldsymbol{S}_{\mathtt{IVW}}(\boldsymbol{\theta})$, the multivariable IVW estimate $\hat{\boldsymbol{\theta}}_{\mathtt{IVW}}$ has approximate bias:

$$\mathrm{E}(\hat{\boldsymbol{\theta}}_{\mathtt{IVW}} - \boldsymbol{\theta}) \approx (\mathbf{B}^\top \mathbf{B} + m \boldsymbol{\Sigma}_{W_\beta W_\beta})^{-1} (m \boldsymbol{\Sigma}_{W_\beta W_\beta} \boldsymbol{\theta} - m \boldsymbol{\sigma}_{W_\beta w_\alpha}), \tag{8}$$

where $\boldsymbol{w}_{\beta_j} = (w_{\beta_{1j}}, \ldots, w_{\beta_{pj}})^\top$, $\boldsymbol{\Sigma}_{W_\beta W_\beta} = \mathrm{cov}(\boldsymbol{w}_{\beta_j})$, and $\sigma_{W_\beta w_\alpha} = \mathrm{cov}(\boldsymbol{w}_{\beta_j}, w_{\alpha_j})$. In (8), $\mathbf{B}^\top \mathbf{B}/m$ can be interpreted as the average information contributed by each IV and $\boldsymbol{\Sigma}_{W_\beta W_\beta}$ can be interpreted as the average information contributed by each estimation error. Therefore, the weak instrument bias will become non-negligible if $\mathbf{B}^\top \mathbf{B}/m$ is not significantly larger than

$\boldsymbol{\Sigma}_{W_\beta W_\beta}$. As the increasing sample sizes of GWAS cohorts, more causal variants with small effect sizes are being identified, increasing the potential for larger m and smaller $\mathbf{B}^\top\mathbf{B}/m$. Lorincz-Comi *et al.* (2023) showed that unless $m/\sqrt{n}_{\min} \to 0$ where $n_{\min}$ is the minimum sample sizes of exposures and outcome GWAS cohorts, it is impossible to make asymptotically valid causal inferences based on $\hat{\boldsymbol{\theta}}_{\texttt{IVW}}$.

2.3. MRBEE-IVW

The original MRBEE method introduced in Lorincz-Comi *et al.* (2023) removes the weak instrument bias of multivariable IVW (noted as MRBEE-IVW). Specifically, MRBEE-IVW estimates $\boldsymbol{\theta}$ by solving the following bias-corrected estimating equation:

$$\boldsymbol{S}_{\texttt{BEE-IVW}}(\boldsymbol{\theta}) = \boldsymbol{S}_{\texttt{IVW}}(\boldsymbol{\theta}) - m(\boldsymbol{\Sigma}_{W_\beta W_\beta}\boldsymbol{\theta} - \boldsymbol{\sigma}_{W_\beta w_\alpha}), \tag{9}$$

where $\boldsymbol{S}_{\texttt{IVW}}(\boldsymbol{\theta}) = -\widehat{\mathbf{B}}^\top(\hat{\boldsymbol{\alpha}} - \widehat{\mathbf{B}}\boldsymbol{\theta})$. The later part in (9) consists of bias attributed to weak instrument and sample overlapping among exposure and outcome cohorts while the horizontal pleiotropy bias is solved through an iterative procedure during solving the equation. The solution $\hat{\boldsymbol{\theta}}_{\texttt{BEE-IVW}}$ such that $\boldsymbol{S}_{\texttt{BEE-IVW}}(\hat{\boldsymbol{\theta}}_{\texttt{BEE-IVW}}) = \mathbf{0}$ is

$$\hat{\boldsymbol{\theta}}_{\texttt{BEE-IVW}} = (\widehat{\mathbf{B}}^\top\widehat{\mathbf{B}} - m\boldsymbol{\Sigma}_{W_\beta W_\beta})^{-1}(\widehat{\mathbf{B}}^\top\hat{\boldsymbol{\alpha}} - m\boldsymbol{\sigma}_{W_\beta w_\alpha}). \tag{10}$$

Lorincz-Comi *et al.* (2023) proved under the moderate condition $m = O(n_{\min})$,

$$\sqrt{n}_{\min}(\hat{\boldsymbol{\theta}}_{\texttt{BEE-IVW}} - \boldsymbol{\theta}) \xrightarrow{D} \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\texttt{BEE-IVW}}(\boldsymbol{\theta}))$$

where $\boldsymbol{\Sigma}_{\texttt{BEE-IVW}}(\boldsymbol{\theta})$ is a constant positive definite symmetric matrix. In other words, MRBEE-IVW can yield unbiased causal effect estimates even as the number of IVs is of the same order as the GWAS sample sizes.

In MRBEE-IVW, we must estimate the bias-correction terms $\boldsymbol{\Sigma}_{W_\beta W_\beta}$ and $\boldsymbol{\sigma}_{W_\beta w_\alpha}$. Here, we follow (Zhu *et al.* 2015) and (Lorincz-Comi *et al.* 2023) and use nonsignificant GWAS summary statistics to estimate them. Let $\hat{\alpha}_j^*, \hat{\beta}_{j1}^*, \ldots, \hat{\beta}_{jp}^*$ $(j = 1, \ldots, M)$ be M GWAS association estimates that have GWAS P-values greater than 0.05 for all exposures and the outcome. Lorincz-Comi *et al.* (2023) showed that $\hat{\alpha}_j^*$ and $\hat{\beta}_{js}^*$ follow the same distributions of w_{α_j} and $w_{\beta_{js}}$, respectively, and hence $\boldsymbol{\Sigma}_{W_\beta \times w_\alpha}$ can be estimated by

$$\widehat{\boldsymbol{\Sigma}}_{W_\beta \times w_\alpha} = \frac{1}{M}\sum_{j=1}^{M}(\hat{\beta}_{j1}^*, \ldots, \hat{\beta}_{jp}^*, \hat{\alpha}_j^*)^\top(\hat{\beta}_{j1}^*, \ldots, \hat{\beta}_{jp}^*, \hat{\alpha}_j^*). \tag{11}$$

In (11), $\widehat{\boldsymbol{\Sigma}}_{W_\beta W_\beta}$ is the first $(p \times p)$ sub-matrix of $\widehat{\boldsymbol{\Sigma}}_{W_\beta \times w_\alpha}$ corresponding to the exposures and $\boldsymbol{\sigma}_{W_\beta w_\alpha}$ contains the first $p-1$ elements that are in the last column of $\widehat{\boldsymbol{\Sigma}}_{W_\beta \times w_\alpha}$. Note that $\widehat{\boldsymbol{\Sigma}}_{W_\beta \times w_\alpha}$ is theoretically a correlation matrix if each GWAS effect size estimate is standardized. An alternative approach is to first estimate the correlation matrix of $(\hat{\beta}_{j1}^*, \ldots, \hat{\beta}_{jp}^*, \hat{\alpha}_j^*)^\top$ then convert it to $\widehat{\boldsymbol{\Sigma}}_{W_\beta \times w_\alpha}$ using the standard errors of $(\hat{\beta}_{j1}^*, \ldots, \hat{\beta}_{jp}^*, \hat{\alpha}_j^*)^\top$ from the GWAS summary data.

We can yield the covariance matrix of $\hat{\boldsymbol{\theta}}_{\texttt{BEE-IVW}}$ using the sandwich formula:

$$\boldsymbol{\Sigma}_{\texttt{BEE-IVW}}(\boldsymbol{\theta}) = \mathbf{F}_{\texttt{BEE-IVW}}^{-1}\mathbf{V}_{\texttt{BEE-IVW}}(\boldsymbol{\theta})\mathbf{F}_{\texttt{BEE-IVW}}^{-1}, \tag{12}$$

where $\mathbf{F}^{-1}_{\texttt{BEE-IVW}}$ is the inverse Fisher information matrix and $\mathbf{V}_{\texttt{BEE-IVW}}(\boldsymbol{\theta})$ is the covariance matrix of $\boldsymbol{S}_{\texttt{BEE-IVW}}(\boldsymbol{\theta})$. We can consistently estimate $\boldsymbol{\Sigma}_{\texttt{BEE-IVW}}(\boldsymbol{\theta})$ using

$$\widehat{\boldsymbol{\Sigma}}_{\texttt{BEE-IVW}}(\hat{\boldsymbol{\theta}}_{\texttt{BEE-IVW}}) = \widehat{\mathbf{F}}^{-1}_{\texttt{BEE-IVW}}\widehat{\mathbf{V}}_{\texttt{BEE-IVW}}(\hat{\boldsymbol{\theta}}_{\texttt{BEE-IVW}})\widehat{\mathbf{F}}^{-1}_{\texttt{BEE-IVW}}, \tag{13}$$

where $\widehat{\mathbf{F}}_{\texttt{BEE-IVW}} = \widehat{\mathbf{B}}^\top\widehat{\mathbf{B}}/m - \widehat{\boldsymbol{\Sigma}}_{W_\beta W_\beta}$, $\widehat{\mathbf{V}}_{\texttt{BEE-IVW}}(\hat{\boldsymbol{\theta}}_{\texttt{BEE-IVW}}) = \sum_{j=1}^m \hat{\boldsymbol{S}}_j(\hat{\boldsymbol{\theta}}_{\texttt{BEE-IVW}})\hat{\boldsymbol{S}}_j(\hat{\boldsymbol{\theta}}_{\texttt{BEE-IVW}})^\top/m$, and $\hat{\boldsymbol{S}}_j(\hat{\boldsymbol{\theta}}_{\texttt{BEE-IVW}}) = -(\hat{\alpha}_j - \hat{\boldsymbol{\theta}}^\top_{\texttt{BEE-IVW}}\hat{\boldsymbol{\beta}}_j)\hat{\boldsymbol{\beta}}_j - \widehat{\boldsymbol{\Sigma}}_{W_\beta W_\beta}\hat{\boldsymbol{\theta}}_{\texttt{BEE-IVW}} + \hat{\boldsymbol{\sigma}}_{W_\beta w_\alpha}$.

2.4. MRBEE-IMRP

As mentioned in the introduction, MRBEE-IVW corrects weak instrument bias in the multivariable IVW estimator which assumes the absence of UHP and CHP captured by $\boldsymbol{\gamma}$ in (4). To make MRBEE-IVW robust to horizontal pleiotropy, we incorporated the IMRP technique (Zhu *et al.* 2021) into MRBEE-IVW to detect horizontal pleiotropy and denote the estimator as MRBEE-IMRP. Specifically, we consider the following IV-specific hypothesis test:

$$\mathbf{H}_0 : \ \gamma_j = 0, \qquad \text{v.s.} \qquad \mathbf{H}_1 : \ \gamma_j \neq 0. \tag{14}$$

Given a current estimate $\boldsymbol{\theta}^{(t)}$, a natural estimate of γ_j is

$$\gamma_j^{(t)} = \hat{\alpha}_j - \hat{\boldsymbol{\beta}}_j^\top\boldsymbol{\theta}^{(t)}, \tag{15}$$

The test statistic of (14) is

$$t^{(t)}_{\gamma_j} = \frac{(\gamma_j^{(t)})^2}{\widehat{\texttt{var}}(\gamma_j^{(t)})}, \tag{16}$$

where $\widehat{\texttt{var}}(\gamma_j^{(t)}) = \boldsymbol{\vartheta}^{(t)^\top}\widehat{\boldsymbol{\Sigma}}_{W_\beta\times w_\alpha}\boldsymbol{\vartheta}^{(t)}$, and $\boldsymbol{\vartheta}^{(t)} = (\boldsymbol{\theta}^{(t)^\top}, -1)^\top$. Then $\gamma_j^{(t)}$ is considered as an outlier if $F_{\chi^2_1}(t^{(t)}_{\gamma_j}) > \kappa$, where $F_{\chi^2_1}(\cdot)$ is the CDF of χ^2_1-distribution and κ is a given threshold. In practice, MRBEE-IMRP iteratively applies the hypothesis test (14) to remove the outliers and uses the remaining IVs to estimate $\hat{\boldsymbol{\theta}}_{\texttt{BEE-IVW}}$. The stable estimate is regarded as $\hat{\boldsymbol{\theta}}_{\texttt{BEE-IMRP}}$. Algorithm 1 provides the pseudo-code of MRBEE-IMRP. Lorincz-Comi *et al.* (2023) proved that the underlying set of IVs with horizontal pleiotropy can be consistently identified if $m/n_{\min} \to 0$.

2.5. MRBEE-IPOD

Most robust MVMR methods such as MR-Robust (Rees *et al.* 2019) and its multivariable version (Grant and Burgess 2021) estimate $\boldsymbol{\theta}$ by minimizing:

$$\hat{\boldsymbol{\theta}}_{\texttt{ROB}} = \arg\min_{\boldsymbol{\theta}} \left\{ \sum_{j=1}^m \rho_\lambda(\hat{\alpha}_j - \hat{\boldsymbol{\beta}}_j^\top\boldsymbol{\theta}) \right\}, \tag{17}$$

where $\rho_\lambda(\cdot)$ is a robust loss function and λ is a tuning parameter determining the degree of robustness (Huber 2011). The score function of (17) is below:

$$\boldsymbol{S}_{\texttt{ROB}}(\boldsymbol{\theta}) = -\sum_{j=1}^m \psi_\lambda(\hat{\alpha}_j - \hat{\boldsymbol{\beta}}_j^\top\boldsymbol{\theta})\hat{\boldsymbol{\beta}}_j, \tag{18}$$

Algorithm 1 Pseudo-code of MRBEE-IMRP.

Input: Initial estimate $\boldsymbol{\theta}^{(0)}$, bias-correction terms $\widehat{\boldsymbol{\Sigma}}_{W_\beta W_\beta}$ and $\hat{\boldsymbol{\sigma}}_{W_\beta w_\alpha}$, hypothesis threshold τ, tolerance ϵ.

while do $||\boldsymbol{\theta}^{(t+1)} - \boldsymbol{\theta}^{(t)}||_2 > \epsilon$

Estimate the pleiotropy $\gamma_j^{(t)}$ by (15),

Calculate the testing statistics $t_{\gamma_j}^{(t)}$ by (16),

Identify the horizontal pleiotropy set $\mathbb{O}^{(t)} = \{j : \ F_{\chi_1^2}(t_{\gamma_j}^{(t)}) > \kappa\}$,

Yield the vertical pleiotropy set $\bar{\mathbb{O}}^{(t)} = \{j : \ F_{\chi_1^2}(t_{\gamma_j}^{(t)}) \leq \kappa\}$,

Update $\boldsymbol{\theta}^{(t+1)}$ by solving (9) with IVs in $\bar{\mathbb{O}}^{(t)}$,

end while

Output: Causal effect estimate $\hat{\boldsymbol{\theta}}_{\texttt{BEE-IMRP}}$, horizontal pleiotropy set $\hat{\mathbb{O}}_{\texttt{BEE-IMRP}}$.

where $\psi_\lambda(\cdot)$ is the derivative of $\rho_\lambda(\cdot)$. The Huber loss function is the most classic robust function, whose expression and derivative are:

$$\rho_\lambda(x) = \begin{cases} \frac{1}{2}x^2, & |x| \leq \lambda \\ \lambda(|x| - \frac{1}{2}\lambda), & |x| > \lambda \end{cases}, \quad \psi_\lambda(x) = \begin{cases} x, & |x| \leq \lambda \\ \lambda \text{sign}(x), & |x| > \lambda \end{cases} \tag{19}$$

Other common robust loss functions include the Tukey's bi-square loss function and Hampels loss function. In the literature, the statistical methods using robust loss functions are called M-estimation (Huber 2011).

Unlike a quadratic loss function whose derivative is linear, it is difficult to obtain the expectation of $\boldsymbol{S}_{\texttt{ROB}}(\boldsymbol{\theta})$ due to the non-linearity of $\psi_\lambda(\cdot)$. Fortunately, recent investigation (She and Owen 2011) revealed that most robust loss functions correspond to a penalized quadratic function with a specific penalty. Specifically, consider the following thresholding function:

$$\Theta_\lambda(x) = \arg\min_t \left\{ \frac{1}{2}(x-t)^2 + p_\lambda(t) \right\}, \tag{20}$$

where $p_\lambda(\cdot)$ is a specific penalty function. She and Owen (2011) proved that if $\Theta_\lambda(x)$ satisfies the thresholding rule defined in their paper, there exists a function $\psi_\lambda(\cdot)$ such that $\forall x$:

$$\Theta_\lambda(x) + \psi_\lambda(x) = x. \tag{21}$$

In other words, $\hat{\boldsymbol{\theta}}_{\texttt{ROB}}$ estimated from (17) is exactly equal to the one yielded by the following joint minimization of causal effect $\boldsymbol{\theta}$ and horizontal pleiotropy $\boldsymbol{\gamma}$:

$$(\hat{\boldsymbol{\theta}}_{\texttt{ROB}}, \hat{\boldsymbol{\gamma}}_{\texttt{ROB}}) = \arg\min_{(\boldsymbol{\theta},\boldsymbol{\gamma})} \left\{ \sum_{j=1}^{m} \left(\frac{1}{2}(\hat{\alpha}_j - \hat{\boldsymbol{\beta}}_j^\top \boldsymbol{\theta} - \gamma_j)^2 + p_\lambda(\gamma_j) \right) \right\}. \tag{22}$$

They further showed that the Huber loss function is equivalent to the penalized quadratic function with lasso penalty $p_\lambda^{\text{lasso}}(x) = \lambda|x|$ (Tibshirani 1996), which means that MR-robust with Huber loss function and MR-lasso (Kang *et al.* 2016) are exactly the same method. In addition, they pointed out that the skipped-mean loss function corresponds to hard peanlty, the Tukey's bisquare loss function corresponds to Tukey penalty, and the Hampels loss function

Algorithm 2 Pseudo-code of MRBEE-IPOD.

Input: Initial estimate $\boldsymbol{\theta}^{(0)}$, bias-correction terms $\widehat{\boldsymbol{\Sigma}}_{W_\beta W_\beta}$ and $\hat{\boldsymbol{\sigma}}_{W_\beta w_\alpha}$, penalty function $p_\lambda(\cdot)$ with a tuning parameter λ, tolerance ϵ.
while do $||\boldsymbol{\theta}^{(t+1)} - \boldsymbol{\theta}^{(t)}||_2 > \epsilon$
Update the horizontal pleiotropy $\boldsymbol{\gamma}^{(t+1)}$ by (25),
Update the causal effect $\boldsymbol{\theta}^{(t+1)}$ by (26),
end while
Output: Causal effect estimate $\hat{\boldsymbol{\theta}}_{\texttt{BEE-IPOD}}$, horizontal pleiotropy estimate $\hat{\boldsymbol{\gamma}}_{\texttt{BEE-IPOD}}$.

corresponds to the smoothly clipped absolute deviation (SCAD) (Fan and Li 2001) penalty:

$$p_\lambda^{\text{SCAD}}(x) = \int_0^{|x|} \lambda\Big\{\mathbf{1}(t \le \lambda) + \frac{(a\lambda - t)_+}{(a-1)\lambda}\mathbf{1}(t > \lambda)\Big\}dt, \tag{23}$$

where $a > 2$ is a tuning parameter.

We prove by using the Karush-Kuhn-Tucker (KKT) condition that the bias-correction version of $\boldsymbol{S}_{\texttt{ROB}}(\boldsymbol{\theta})$ is

$$\boldsymbol{S}_{\texttt{BEE-ROB}}(\boldsymbol{\theta}) = \boldsymbol{S}_{\texttt{ROB}}(\boldsymbol{\theta}) - \hat{v}_{\texttt{ROB}}(\boldsymbol{\Sigma}_{W_\beta W_\beta}\boldsymbol{\theta} - \boldsymbol{\sigma}_{W_\beta w_\alpha}), \tag{24}$$

where $\hat{v}_{\texttt{ROB}} = \sum_{j=1}^m \mathbf{1}(\hat{\gamma}_{j\texttt{ROB}} = 0)$ is the number of IVs without horizontal pleiotropy evidence. In other words, if we simultaneously estimate the causal effect $\boldsymbol{\theta}$ and the horizontal pleiotropy $\boldsymbol{\gamma}$ with (22), only the IVs without horizontal pleiotropy evidence will contribute to weak instrument bias. We apply the iterative procedure for outlier detection (IPOD) procedure (She and Owen 2011) to update $\boldsymbol{\theta}$ and $\boldsymbol{\gamma}$: given the current estimate $\boldsymbol{\theta}^{(t+1)}$, $\boldsymbol{\gamma}^{(t)}$ is updated through:

$$\boldsymbol{\gamma}^{(t+1)} = \Theta_\lambda(\hat{\boldsymbol{\alpha}} - \widehat{\mathbf{B}}\boldsymbol{\theta}^{(t)}), \tag{25}$$

given the current estimate $\boldsymbol{\theta}^{(t+1)}$, $\boldsymbol{\gamma}^{(t+1)}$ is updated through:

$$\boldsymbol{\theta}^{(t+1)} = (\widehat{\mathbf{B}}^\top\widehat{\mathbf{B}} - v^{(t+1)}\widehat{\boldsymbol{\Sigma}}_{W_\beta W_\beta})^{-1}(\widehat{\mathbf{B}}^\top(\hat{\boldsymbol{\alpha}} - \boldsymbol{\gamma}^{(t+1)}) - v^{(t+1)}\hat{\boldsymbol{\sigma}}_{W_\beta w_\alpha}), \tag{26}$$

where $v^{(t+1)} = \sum_{j=1}^m \mathbf{1}(\gamma_j^{(t+1)} = 0)$. The stable solutions of $\boldsymbol{\theta}$ and $\boldsymbol{\gamma}$ are considered the outputs of IPOD procedure. We name this novel procedure to remove weak instrumental bias of robust MVMR methods as MRBEE-IPOD, and denote the stable solutions as $\hat{\boldsymbol{\theta}}_{\texttt{BEE-IPOD}}$ and $\hat{\boldsymbol{\gamma}}_{\texttt{BEE-IPOD}}$. Algorithm 2 provides the pseudo-code of MRBEE-IPOD.

2.6. MRBEE-Median

The MR-Median and its multivariable version (Bowden *et al.* 2016) can be regarded as an alternative robust method that estimates $\boldsymbol{\theta}$ by

$$\hat{\boldsymbol{\theta}}_{\texttt{MED}} = \arg\min_{\boldsymbol{\theta}} \Big\{\sum_{j=1}^m q_{\frac{1}{2}}(\hat{\alpha}_j - \hat{\boldsymbol{\beta}}_j^\top\boldsymbol{\theta})\Big\} \tag{27}$$

where $q_\tau(x) = x(\tau - \mathbf{1}(x < 0))$ is the quantile loss function which reduces to absolute value function if $\tau = 0.5$. Similar to a classical robust loss function, it is challenging to derive the

expectation of the estimating function of (27) since the quantile loss function $q_\tau(x)$ is not differentiable at $x = 0$. Also, the quantile loss function does not satisfy the thresholding rule and hence cannot be handled through the MRBEE-IPOD procedure.

Wang *et al.* (2012) proposed bias-correction for measurement error in quantile regression. Specifically, they considered an approximation of the quantile loss function:

$$q_\tau(x, h) = x(\tau - 0.5 + \mathrm{Si}(x/h)) \tag{28}$$

where $\mathrm{Si}(x) = \pi^{-1} \int_0^x \sin(t)dt$ is the sine integral kernel function and h is a bandwidth. Then they pointed out that if $\epsilon \sim \mathcal{N}(0, \sigma^2)$ and $u \sim \mathcal{N}(0, 1)$ being independent of ϵ, $\mathrm{E}(\mathrm{E}(g(\epsilon + i\sigma u|\epsilon))) = g(u)$ with $i = \sqrt{-1}$ (Stefanski and Cook 1995). Using this conclusion, they proposed the following unbiased approximation of quantile loss function

$$q_\tau(x, h, \sigma^2) = x(\tau - 0.5) + \pi^{-1} \int_0^{\frac{1}{h}} (t^{-1} x \sin(tx) - \sigma^2 \cos(tx) \exp(t^2\sigma^2/2)dt), \tag{29}$$

where σ^2 is the variance of measurement error multiplied by regression coefficient.

Inspired by this result, we propose a novel MVMR-Median method named as MRBEE-Median, which estimates $\boldsymbol{\theta}$ by solving

$$\hat{\boldsymbol{\theta}}_{\texttt{BEE-MED}} = \arg\min_{\boldsymbol{\theta}} \left\{ q_{\frac{1}{2}}(\hat{\alpha}_j - \hat{\boldsymbol{\beta}}_j^\top \boldsymbol{\theta}, h, \sigma^2) \right\} \tag{30}$$

where under our standardization σ^2 is equal to

$$\sigma^2 = \mathtt{var}(w_{\alpha_j} - \boldsymbol{W}_{\beta_j}^\top \boldsymbol{\theta}) = 1 + \boldsymbol{\theta}^\top \boldsymbol{\Sigma}_{W_\beta W_\beta} \boldsymbol{\theta} - 2\boldsymbol{\sigma}_{W_\beta w_\alpha}^\top \boldsymbol{\theta}. \tag{31}$$

As numerical integration is unavoidable in $q_\tau(x, h, \sigma^2)$, we directly employ the R function `optim()` with the L-BFGS-B algorithm (Liu and Nocedal 1989) to minimize (30). The output of `optim()` is considered the solution to (30) and is denoted as $\hat{\boldsymbol{\theta}}_{\texttt{BEE-MED}}$. The variance-covariance matrix of $\hat{\boldsymbol{\theta}}_{\texttt{BEE-MED}}$ is taken to be the Hessian matrix of (27) that is estimated by numerical optimization with the L-BFGS-B algorithm.

2.7. MRBEE-MLqE

In the following two subsections, we introduce two new MVMR methods as alternative options to MRBEE-IMRP, MRBEE-IPOD, and MRBEE-Median we previously proposed. The first approach involves reducing the influence of IVs with significant horizontal pleiotropy evidence by assigning them smaller weights compared to those with vertical pleiotropy. The bias-correction terms for this approach are identical to those of MRBEE-IVW because the reweighting scheme does not change the linearity of MRBEE-IVW. We propose a new method called MRBEE-MLqE, which combines the Maximum Lq-likelihood estimate (Ferrari and Yang 2010) with MRBEE-IVW. The estimating equation for MRBEE-MLqE is:

$$\boldsymbol{S}_{\texttt{BEE-MLqE}}(\boldsymbol{\theta}) = \sum_{j=1}^{m} \omega_j \left\{ (\hat{\boldsymbol{\beta}}_j^\top \boldsymbol{\theta} - \hat{\alpha}_j)\hat{\boldsymbol{\beta}}_j - \boldsymbol{\Sigma}_{W_\beta W_\beta} \boldsymbol{\theta} + \boldsymbol{\sigma}_{W_\beta w_\alpha} \right\}, \tag{32}$$

where the weight ω_j is equal to

$$\omega_j = f(\hat{\boldsymbol{\beta}}_j^\top \boldsymbol{\theta} - \hat{\alpha}_j; 0, \sigma^2)^{1-q}, \tag{33}$$

Algorithm 3 Pseudo-code of MRBEE-MLqE.

Input: Initial estimate $\boldsymbol{\theta}^{(0)}$, bias-correction terms $\widehat{\boldsymbol{\Sigma}}_{W_\beta W_\beta}$ and $\hat{\boldsymbol{\sigma}}_{W_\beta w_\alpha}$, reweighting parameter q, tolerance ϵ.
while do $||\boldsymbol{\theta}^{(t+1)} - \boldsymbol{\theta}^{(t)}||_2 > \epsilon$
Update the weights $\{\omega_j^{(t+1)}\}$ by (35),
Update the causal effect $\boldsymbol{\theta}^{(t+1)}$ by (36),
end while
Output: Causal effect estimate $\hat{\boldsymbol{\theta}}_{\texttt{BEE-MLqE}}$, weights of IVs $\{\hat{\omega}_{j\texttt{BEE-MLqE}}\}$.

where $\sigma^2 = \texttt{var}(\hat{\boldsymbol{\beta}}_j^\top \boldsymbol{\theta} - \hat{\alpha}_j)$, $q \in (0, 1)$ is a reweighting parameter, and $f(x; \mu, \sigma^2)$ is the probability density function (PDF) of univariable normal distribution

$$f(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\Big(- \frac{(x-\mu)^2}{2\sigma^2}\Big). \tag{34}$$

It is easy to see that MRBEE-MLqE assigns weights to IVs based on the $(1-q)$ power of the normal probability density function, which will reduce the contributions of IVs with larger pleiotropic evidence, which can rapidly approach zero because the normal probability density function decreases exponentially. When $q = 1$, MRBEE-MLqE is equivalent to MRBEE-IVW, which uses a quadratic function as the loss function. A more detailed introduction of MLqE is shown in (Ferrari and Yang 2010).

This approach requires weights which depend on causal estimates. As a result, we must again apply an iterative procedure to arrive at stable causal estimates. Given a current estimate $\boldsymbol{\theta}^{(t)}$, we update the weight

$$\omega_j^{(t+1)} = f(\hat{\boldsymbol{\beta}}_j^\top \boldsymbol{\theta}^{(t)} - \hat{\alpha}_j; 0, \sigma^2)^{1-q} \tag{35}$$

where $\sigma^{(t)2} = \boldsymbol{\vartheta}^{(t)\top} \widehat{\boldsymbol{\Sigma}}_{W_\beta \times w_\alpha} \boldsymbol{\vartheta}^{(t)}$, $\boldsymbol{\vartheta}^{(t)} = (\boldsymbol{\theta}^{(t)\top}, -1)^\top$, and we standardize the weights such that $\sum_{j=1}^m \omega_j^{(t+1)} = 1$ for convenience. Based on the current weights $\{\omega_j^{(t)}\}$, we update $\boldsymbol{\theta}$ by

$$\boldsymbol{\theta}^{(t+1)} = (\widehat{\mathbf{B}}^\top \boldsymbol{\Omega}^{(t)} \widehat{\mathbf{B}} - \widehat{\boldsymbol{\Sigma}}_{W_\beta W_\beta})^{-1} (\widehat{\mathbf{B}}^\top \boldsymbol{\Omega}^{(t)} \hat{\boldsymbol{\alpha}} - \hat{\boldsymbol{\sigma}}_{W_\beta w_\alpha}), \tag{36}$$

where $\boldsymbol{\Omega}^{(t)}$ is a diagonal matrix with the jth diagonal entry $\omega_j^{(t)}$. The stable solutions of $\boldsymbol{\theta}$ and $\{\omega_j\}$ are considered as the outputs of MRBEE-MLqE and are denoted as $\hat{\boldsymbol{\theta}}_{\texttt{BEE-MLqE}}$ and $\{\hat{\omega}_{j\texttt{BEE-MLqE}}\}$. Algorithm 3 provides the pseudo-code of MRBEE-MLqE.

2.8. MRBEE-Mixture

If causal effects are estimated using different IV clusters, it may indicate the presence of multiple causal mechanisms by which the exposure affects the outcome (Burgess *et al.* 2023). While most MR methods, including those presented above, consider this phenomenon to be horizontal pleiotropy and therefore avoid it, we can also characterize the alternative causal pathways by assuming a mixture of causal effects among the set of IVs. Our novel approach, named MRBEE-Mixture, is designed for this purpose. Specifically, MRBEE-Mixture considers a two-cluster model:

$$\hat{\alpha}_j = \hat{\boldsymbol{\beta}}_j^\top \Big(\delta_j \boldsymbol{\theta}_1 + (1 - \delta_j) \boldsymbol{\theta}_2 \Big) + \varepsilon_j. \tag{37}$$

where δ_j is a binary variable with frequency π, $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ are the two causal effect vectors of the two causal pathways, and ϵ_j is the residual following the distributions:

$$\epsilon_j \sim \begin{cases} \mathcal{N}(0, 1 + \boldsymbol{\theta}_1^\top \boldsymbol{\Sigma}_{W_\beta W_\beta} \boldsymbol{\theta}_1 - 2\boldsymbol{\sigma}_{W_\beta w_\alpha}^\top \boldsymbol{\theta}_1), & \delta_j = 1, \\ \mathcal{N}(0, 1 + \boldsymbol{\theta}_2^\top \boldsymbol{\Sigma}_{W_\beta W_\beta} \boldsymbol{\theta}_2 - 2\boldsymbol{\sigma}_{W_\beta w_\alpha}^\top \boldsymbol{\theta}_2), & \delta_j = 0. \end{cases} \tag{38}$$

Here, δ_j indicates if the jth IV belongs to the first cluster, and π is the fraction of IVs belonging to the first cluster. While it is technically feasible to expand MRBEE-Mixture to encompass more than two sub-clusters, we limit our focus to two sub-clusters because it is less probable for the number of sub-clusters to exceed two.

MRBEE-Mixture employs an alternative reweighting scheme that is applied to cluster the IVs and estimate the causal effects simultaneously, where the unbiased estimating equations are

$$\boldsymbol{S}_{\texttt{BEE-MIX}}(\boldsymbol{\theta}_1) = \sum_{j=1}^{m} \delta_j \Big\{ (\hat{\boldsymbol{\beta}}_j^\top \boldsymbol{\theta}_1 - \hat{\alpha}_j)\hat{\boldsymbol{\beta}}_j - \boldsymbol{\Sigma}_{W_\beta W_\beta} \boldsymbol{\theta}_1 + \boldsymbol{\sigma}_{W_\beta w_\alpha} \Big\}, \tag{39}$$

$$\boldsymbol{S}_{\texttt{BEE-MIX}}(\boldsymbol{\theta}_2) = \sum_{j=1}^{m} (1-\delta_j) \Big\{ (\hat{\boldsymbol{\beta}}_j^\top \boldsymbol{\theta}_2 - \hat{\alpha}_j)\hat{\boldsymbol{\beta}}_j - \boldsymbol{\Sigma}_{W_\beta W_\beta} \boldsymbol{\theta}_2 + \boldsymbol{\sigma}_{W_\beta w_\alpha} \Big\}. \tag{40}$$

Given two current causal effect estimates $\boldsymbol{\theta}_1^{(t)}$ and $\boldsymbol{\theta}_2^{(t)}$, the indicator is updated by

$$\delta_j^{(t+1)} = \frac{f(\hat{\boldsymbol{\beta}}_j^\top \boldsymbol{\theta}_1^{(t)} - \hat{\alpha}_j; 0, \sigma_1^{(t)2})}{f(\hat{\boldsymbol{\beta}}_j^\top \boldsymbol{\theta}_1^{(t)} - \hat{\alpha}_j; 0, \sigma_1^{(t)2}) + f(\hat{\boldsymbol{\beta}}_j^\top \boldsymbol{\theta}_2^{(t)} - \hat{\alpha}_j; 0, \sigma_2^{(t)2})}, \tag{41}$$

where $f(x; \mu, \sigma^2)$ is the PDF of univariable normal distribution and

$$\sigma_k^{(t)2} = 1 + \boldsymbol{\theta}_k^{(t)\top} \hat{\boldsymbol{\Sigma}}_{W_\beta W_\beta} \boldsymbol{\theta}_k^{(t)} - 2\hat{\boldsymbol{\sigma}}_{W_\beta w_\alpha}^\top \boldsymbol{\theta}_k^{(t)}, \quad k = 1, 2. \tag{42}$$

We call $\delta_j^{(t+1)}$ the soft indicator because it is indeed the probability that the jth IV belongs to the first cluster conditional on the current estimates $\boldsymbol{\theta}_1^{(t)}$ and $\boldsymbol{\theta}_2^{(t)}$. Next, given the soft indicator $\{\delta_j^{(t+1)}\}$, the causal effect estimates are updated by

$$\boldsymbol{\theta}_1^{(t+1)} = \Big\{ \sum_{j=1}^{m} \delta_j^{(t+1)} \Big(\hat{\boldsymbol{\beta}}_j \hat{\boldsymbol{\beta}}_j^\top - \hat{\boldsymbol{\Sigma}}_{W_\beta W_\beta} \Big) \Big\}^{-1} \Big\{ \sum_{j=1}^{m} \delta_j^{(t+1)} \Big(\hat{\alpha}_j \hat{\boldsymbol{\beta}}_j - \hat{\boldsymbol{\sigma}}_{W_\beta w_\alpha} \Big) \Big\}, \tag{43}$$

$$\boldsymbol{\theta}_2^{(t+1)} = \Big\{ \sum_{j=1}^{m} (1-\delta_j^{(t+1)}) \Big(\hat{\boldsymbol{\beta}}_j \hat{\boldsymbol{\beta}}_j^\top - \hat{\boldsymbol{\Sigma}}_{W_\beta W_\beta} \Big) \Big\}^{-1} \Big\{ \sum_{j=1}^{m} (1-\delta_j^{(t+1)}) \Big(\hat{\alpha}_j \hat{\boldsymbol{\beta}}_j - \hat{\boldsymbol{\sigma}}_{W_\beta w_\alpha} \Big) \Big\}. \tag{44}$$

Additionally, we cannot rule out the extension of IVs with residual horizontal pleiotropy that neither belong to the first cluster nor the second. To identify these IVs, we perform the following hypothesis test

$$\mathbf{H}_0 : \ \gamma_{1j} = 0 \text{ or } \gamma_{2j} = 0 \quad v.s. \quad \mathbf{H}_0 : \ \gamma_{1j} \neq 0 \text{ and } \gamma_{2j} \neq 0 \tag{45}$$

where $\gamma_{1j} = \alpha_j - \boldsymbol{\beta}^\top \boldsymbol{\theta}_1$ and $\gamma_{2j} = \alpha_j - \boldsymbol{\beta}^\top \boldsymbol{\theta}_2$. The related natural estimates are

$$\gamma_{1j}^{(t+1)} = \hat{\alpha}_j - \hat{\boldsymbol{\beta}}^\top \boldsymbol{\theta}_1^{(t+1)}, \quad \gamma_{2j}^{(t+1)} = \hat{\alpha}_j - \hat{\boldsymbol{\beta}}^\top \boldsymbol{\theta}_2^{(t+1)}, \tag{46}$$

Algorithm 4 Pseudo-code of MRBEE-Mixture.

Input: Initial estimates $\boldsymbol{\theta}_1^{(0)}$ and $\boldsymbol{\theta}_2^{(0)}$, bias-correction terms $\widehat{\boldsymbol{\Sigma}}_{W_\beta W_\beta}$ and $\hat{\boldsymbol{\sigma}}_{W_\beta w_\alpha}$, IMRP status, threshold κ, tolerance ϵ.

while $||\boldsymbol{\theta}_1^{(t+1)} - \boldsymbol{\theta}_1^{(t)}||_2 > \epsilon$ **do**

Update the indicators $\{\delta_j^{(t+1)}\}$ by (41),

Update the causal effects $\boldsymbol{\theta}_1^{(t+1)}$ and $\boldsymbol{\theta}_2^{(t+1)}$ by (43) and (44),

if IMRP=TRUE **then**

Identify the horizontal pleiotropy according to (45),

end if

end while

Output: Causal effect estimates $\hat{\boldsymbol{\theta}}_{\texttt{1BEE-MIX}}$, $\hat{\boldsymbol{\theta}}_{\texttt{2BEE-MIX}}$, soft indicators of IVs $\{\hat{\delta}_{j\texttt{BEE-MIX}}\}$.

and their test statistics are

$$t_{1j}^{(t+1)} = \left(\frac{\gamma_{1j}^{(t+1)}}{\sigma_1^{(t)}}\right)^2, \quad t_{2j}^{(t+1)} = \left(\frac{\gamma_{2j}^{(t+1)}}{\sigma_2^{(t)}}\right)^2, \tag{47}$$

implying that the p-values of the hypothesis test (45) are

$$p^{(t+1)} = \max(1 - F_{\chi_1^2}(t_{1j}^{(t+1)}), 1 - F_{\chi_1^2}(t_{2j}^{(t+1)})). \tag{48}$$

Given a threshold κ, the IVs with p-values smaller than $1 - F_{\chi_1^2}(\kappa)$ are removed in the next iteration. The stable solutions of $\boldsymbol{\theta}_1$, $\boldsymbol{\theta}_2$ and $\{\delta_j\}$ are the outputs of MRBEE-Mixture and are denoted as $\hat{\boldsymbol{\theta}}_{\texttt{1BEE-MIX}}$, $\hat{\boldsymbol{\theta}}_{\texttt{2BEE-MIX}}$, and $\{\hat{\delta}_{j\texttt{BEE-MIX}}\}$. Algorithm 4 provides the pseudo-code of MRBEE-Mixture.

3. Package structure

We now introduce the **MRBEE R** package and demonstrate its usage with real data. The **MRBEE R** package can be used to perform three primary tasks: (i) calculate bias-correction terms, (ii) perform MR using MRBEE and each of the methods introduced above, (iii) investigate horizontal pleiotropy in MR at genome-wide level. The essential functions to perform these tasks are displayed as a workflow in Figure 2. MR typically requires completing a number of data pre-processing steps, including merging multiple GWAS summary statistic data sets, allele harmonization between different GWAS, IV selection and pruning, standardization of IV association estimates, and others. We note here that the **MendelianRandomization R** already performs each of these tasks and users may use those for any the pre-processing that our software does not provide. Our software assumes that the IV set has already been identified and that a single data set of all GWAS summary statistic files has been created. We also assume all GWAS summary statistics have been standardized using the methods that are most appropriate given the data. This is the starting point of MR using the **MRBEE** software. The **MRBEE** software will perform allele harmonization for different GWAS.

In all examples that follow, we assign to the `cardioData` object the merged data frame of all necessary summary information from GWAS for each exposure and outcome. For each exposure and outcome, the following information is required: standardized effect size

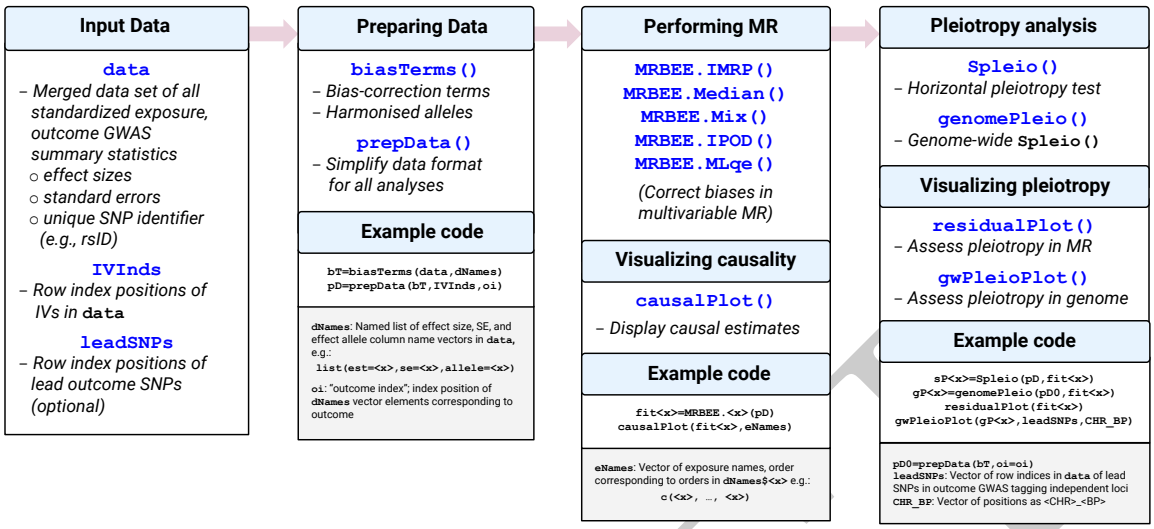

Figure 2: Workflow used to perform MR and horizontal pleiotropy analyses using the **R** package **MRBEE**.

estimates and their standard errors, a unique SNP identifier (e.g., rsID), and effect allele designations. The **MRBEE** package can be downloaded from the github repository https://github.com/noahlorinczcomi/MRBEE using either of the following two commands in R:

```
R> remotes::install_github("noahlorinczcomi/MRBEE")
R> devtools::install_github("noahlorinczcomi/MRBEE")
```

4. Data manipulation

In what follows, we assume that the goal is to estimate the causal effects $\boldsymbol{\theta}$ of 9 cardiometabolic exposures on coronary artery disease (CAD) risk using MRBEE and GWAS summary statistics. The **MRBEE** package comes with a large dataset of merged standardized GWAS summary statistics for these 9 exposures and CAD. These data are stored in the file `cardioData.txt.gz` which can be downloaded from https://www.dropbox.com/s/88fnb1zz0ox2265/cardioData.txt.gz?dl=0. When saved into the current working directory in an **R** session, we can load in these data using the following command:

```
R> library(MRBEE)
R> cardioData=data.table::fread("cardioData.txt.gz")
R> colnames(cardioData)
[1] "rsID" "CHR_BP" "BETA_CAD" "BETA_HDL" "BETA_LDL" "BETA_TG" "BETA_SBP"
[2] "BETA_UA" "BETA_HBA1C" "BETA_BMI" "BETA_HEIGHT" "BETA_HG" "SE_CAD"
[3] "SE_HDL" "SE_LDL" "SE_TG" "SE_SBP" "SE_UA" "SE_HBA1C" "SE_BMI"
[4] "SE_HEIGHT" "SE_HG" "EFFECT_ALLELE_CAD" "EFFECT_ALLELE_HDL"
[5] "EFFECT_ALLELE_LDL" "EFFECT_ALLELE_TG" "EFFECT_ALLELE_SBP"
[6] "EFFECT_ALLELE_UA" "EFFECT_ALLELE_HBA1C" "EFFECT_ALLELE_BMI"
[7] "EFFECT_ALLELE_HEIGHT" "EFFECT_ALLELE_HG" "NONEFFECT_ALLELE_CAD"
```

```
[8] "NONEFFECT_ALLELE_HDL" "NONEFFECT_ALLELE_LDL" "NONEFFECT_ALLELE_TG"
[9] "NONEFFECT_ALLELE_SBP" "NONEFFECT_ALLELE_UA" "NONEFFECT_ALLELE_HBA1C"
[10] "NONEFFECT_ALLELE_BMI" "NONEFFECT_ALLELE_HEIGHT" "NONEFFECT_ALLELE_HG"
[11] "MAF_CAD"
```

To begin MR with **MRBEE**, the user is required to indicate the names of columns in which standardized GWAS estimates, their standardized standard errors, and corresponding effect alleles are present in `cardioData`. The user will create 3 vectors of character elements and place them into a list. The first vector will be a string of column names representing standardized GWAS estimates for CAD and the 9 exposures in `cardioData`, in that order (by choice, not necessity). We can create this vector using the commands:

```
R> exposures=c("HDL","LDL","TG","SBP","UA","HBA1C","BMI","HEIGHT","HG")
R> ests=c("BETA_CAD",paste0("BETA_",exposures"))
```

It is apparent that a well-specified naming convention for summary statistics from the multiple exposure GWAS merged in `cardioData` is extremely helpful. We can then create the second vector of column names representing the standard errors corresponding to `ests` using the command:

```
R> ses=c("SE_CAD",paste0("SE_",exposures))
```

Finally, we can create the third vector of effect allele column names using the command:

```
R> alleles=c("EFFECT_ALLELE_CAD",paste0("EFFECT_ALLELE_", exposures))
```

It is required that the kth index position of `ests` corresponds to the same phenotype as the kth index positions of `ses` and `alleles`. A column naming convention such as the one above of the format `BETA_<x>`, `SE_<x>`, and `EFFECT_ALLELE_<x>` ensures that this requirement is met, although the user is free to choose any naming convention so long as `ests`, `ses`, and `alleles` are specified with the correct orderings of their elements. The next step is to simply place each of these three vectors into a single list called `dNames` using the following command

```
R> dNames=list(est=ests,se=ses,allele=alleles)
```

We will use `dNames` in the subsequent steps to prepare GWAS data for MR. The key names of the elements in the list `dNames` do not need to match those displayed above, but elements in the first position must correspond to effect size estimates, the second to their standard errors, and the third to the effect alleles.

4.1. Bias-correction terms

All robust MRBEE estimators that were described above make a bias-correction to their respective traditional, biased estimators. Calculations of the quantities necessary to make these bias-corrections can be done quickly and easily using the `biasTerms()` function. This function also performs harmonization of GWAS association estimates by effect allele to a user-specified reference phenotype. Allele harmonization is a required step of MR to ensure that GWAS association estimates correspond to increases in copies of the same allele for all phenotypes considered. Given the objects `cardioData` and `dNames`, below is an example of how to use

the `biasTerms()` function to harmonise alleles and calculate $\mathbf{R}_{W_\beta \times w_\alpha} := \text{Corr}([\mathbf{w}_{\beta_j}, w_{\alpha_j}]^\top)$. In `cardioData`, we have already performed the Z-score standardization as in Lorincz-Comi *et al.* (2023) which means that $\mathbf{R}_{W_\beta \times w_\alpha} = \mathbf{\Sigma}_{W_\beta \times w_\alpha}$ from (11). This is because all GWAS effect size estimates are Z-statistics and their standard errors are each 1.

```
R> bT=biasTerms(cardioData,dNames)
R> names(bT)
[1] "R" "Ncor" "EstHarm" "SEHarm"
```

The output of `biasTerms()` assigned to `bT` contains the objects `R` (using our standardization this is equal to $\mathbf{\Sigma}_{W_\beta \times w_\alpha}$), `Ncor` (the number of SNPs used to calculate `R`; 3,112,073 for `cardioData`), `EstHarm` (a matrix of GWAS estimates for the outcome and all exposures harmonized to the phenotype corresponding to the first positions of the vectors `ests`, `ses`, and `alleles` from above), and `SEHarm` (a matrix of standard errors for the outcome and all exposures corresponding to estimates in `EstHarm`). `EstHarm` and `SEHarm` have rows corresponding to the rows of `cardioData` such that the sth row of `cbind(EstHarm,SEHarm)` corresponds to the sth row of `cardioData`. Similarly, the kth column of `EstHarm` corresponds to the kth column of `SEHarm` and the kth row and column of `R`. Since column names of GWAS statistics corresponding to CAD were placed in the first positions of the vectors in `dNames`, the output `EstHarm` and `SEHarm` will have GWAS statistics corresponding to CAD in their first columns. The same is correspondingly the same for the ordering of the exposures. `biasTerms()` will also rename the columns of `SEHarm` to match those of `EstHarm` to remove ambiguity in the column-wise correspondence between `EstHarm` and `SEHarm` that may arise if the user re-orders any of these columns later.

We assume that the index positions of IVs to be used in MR have already been identified in `cardioData` and are stored in the numeric vector `IVInds`. The vector `IVInds` will be used to subset the output `bT` from `biasTerms()` to only those SNPs that will be used to perform MR. Unique SNP identifiers for IVs for many traits can be downloaded from `https://gwas.mrcieu.ac.uk` and their index positions found in `cardioData` very easily using the `cardioData$rsID` column. In our example, the list of rsIDs for the IVs can be downloaded and their index positions in `cardioData` identified using the following commands from the working directory:

```
~$ wget -O IVInds.txt "https://www.dropbox.com/s/r3ppshtrw2ibfy3/IVInds.txt?dl=0"
R> ivs=data.table::fread("IVInds.txt")$SNP # vector of IV rsIDs
R> IVInds=which(cardioData$rsID %in% ivs) # IV index positions in cardioData
```

The next step is to put the data into a simplified format for analysis using the `prepData()` function. To do this, we require one small piece of additional information. The user must specify the index position of elements of `ests`, `ses`, and `alleles` that correspond to the outcome. In our example above, and by default, the correct index position is 1. That is, column names corresponding to CAD-specific variables were first in `ests`, `ses`, and `alleles`. This index position is specified in `prepData()` using the `oi` argument in the following way:

```
R> oi=1 # index position of ests, ses, alleles values corresponding to outcome
R> pD=prepData(bT,IVInds,oi=oi)
R> names(pD)
[1] "betaX" "betaY" "UU" "UV" "VV"
```

The output of `prepData()` (`pD`) is a list of length 5 with the corresponding elements: `pD$betaX` (a matrix of standardized exposure effect sizes), `pD$betaY` (a vector of standardized outcome effect sizes), `pD$UU` (a three-dimensional array of size $p \times p \times m$ where each jth of m elements corresponds to the $p \times p$ submatrix $\boldsymbol{\Sigma}_{W_{\beta_j} \times W_{\beta_j}}$ from (11), `pD$UV` (a three-dimensional array of size $p \times 1 \times m$ where each jth of m elements corresponds to the $p \times 1$ submatrix $\boldsymbol{\sigma}_{W_{\beta_j} \times w_{\alpha_j}}$ from (11), and `pD$VV` (a three-dimensional array of size $1 \times 1 \times m$ where each jth of m elements corresponds to the 1×1 scalar submatrix $\sigma^2_{w_{\alpha_j}}$. At this step, we have converted the measurement error correlation matrix $\mathbf{R}_{W_\beta \times w_\alpha}$ to the m measurement error variance-covariance matrices $\boldsymbol{\Sigma}_{W_{\beta_j} \times w_{\alpha_j}}$ indexed by j for the jth IV. The elements `pD$UU` and `pD$UV` contain values that are the $\boldsymbol{\sigma}_{W_{\beta_j} \times w_{\alpha_j}}$ sub-matrices of $\boldsymbol{\Sigma}_{W_{\beta_j} \times w_{\alpha_j}}$ and are used to correct for bias from weak instruments, sample overlap, and measurement error. This concludes the preparation of data for analysis by **MRBEE**.

5. Mendelian Randomization and pleiotropy analysis

5.1. Performing MR

We now have a family of robust estimators of causal effects using MR that corrects for bias from weak instruments, measurement error, sample overlap, and horizontal pleiotropy. We have made bias-corrections to many popular MR estimators to produce MRBEE-IMRP, MRBEE-Median, MRBEE-Mix, MRBEE-IPOD, and MRBEE-MLqe. Subsequently, we have the corresponding functions `MRBEE.IMRP()`, `MRBEE.Median()`, `MRBEE.Mix()`, `MRBEE.IPOD()`, and `MRBEE.MLqe()` to estimate causal effects using each of these methods. Each of these functions receives the same input, which is the output of `prepData()` (assigned to `pD` from above in our examples to follow) and any additional method-specific arguments if the user wishes to change their defaults. Below is a demonstration of how to estimate causal effects using each of these methods.

```
R> m=nrow(pD$betaX) # counting number of IVs to use in MR
R> debiasedIMRP=MRBEE.IMRP(pD,PleioPThreshold=0.05/sqrt(m)) # 16 seconds
R> debiasedMedian=MRBEE.Median(pD) # 1.1 mins
R> debiasedIPOD=MRBEE.IPOD(pD) # 0.1 secs
R> debiasedMix=MRBEE.Mix(pD) # 1 secs
R> debiasedML=MRBEE.MLqe(pD) # <0.1 secs
```

For completeness, assume we have estimated causal effects using the IVW estimator like this:

```
R> pDIVW=pD; pDIVW$UU=pDIVW$UU*0; pDIVW$UV=pDIVW$UV*0 # ignoring bias-correction
R> IVW=MRBEE.IMRP(pDIVW,PleioPThreshold=0)  # ignoring horizontal pleiotropy
```

which sets the bias-correction terms to be matrices of 0s which will ensure they have no impact on causal estimation, reducing the output of `IVW` to results that would be achieved by using the standard IVW estimator. The objects to which the outputs of the commands `MRBEE.<x>` are assigned are lists of varying length that provide causal estimates, standard errors, and method-specific diagnostic and performance metrics. List entries corresponding to causal estimates have the names `$CausalEstimates` for all methods. Standard errors can be extracted from a

fitted object named (e.g.) `fit` using `sqrt(diag(fit$VCovCausalEstimates))` for all methods. We can view causal estimates made by each of these methods using the `causalPlot()` function. We can either give `causalPlot()` a single fitted model object (e.g., `debiasedIPOD`), or a named list of multiple fitted model objects like this:

```
R> fitList=list(
+   "IVW"=IVW,
+   "MRBEE-IMRP"=debiasedIMRP,
+   "MRBEE-Median"=debiasedMedian,
+   "MRBEE-IPOD"=debiasedIPOD,
+   "MRBEE-Mixture"=debiasedMix,
+   "MRBEE-ML"=debiasedML).
```

We will use the object `fitList` to display causal estimates from multiple methods together using the `causalPlot()` function. For plotting purposes, we may wish to use more readable exposure names in plotting than were originally available in the column names of `cardioData`. To do this, we can create the vector `cleaner_exposure_names` like this

```
R> cleaner_exposure_names=c("HDL","LDL","TG","SBP","Uric acid",
+   "HbA1c","BMI","Height","Hemoglobin").
```

Note that this is not required to create plots using `causalPlot()`, but it may make the appearance of the plot better. We can view estimates of the causal effects of multiple exposure on CAD using multiple MR methods using the following command:

```
R> causalPlot(fitList,cleaner_exposure_names,binaryOutcome=TRUE)+
+   labs(y="Odds Ratio (95% CI)")
```

which produces the plot in Figure 3. Figures produced by the `causalPlot()` function are `ggplot` objects from the **ggplot2 R** package and so can be easily modified after using `causalPlot()`, as was demonstrated in the code above when we added new y-axis labels using the `labs()` function. For all methods using the function naming convention `MRBEE.<x>`, P-values for causal estimates can be extracted easily. We demonstrate how to do this for the `MRBEE.Mix` function as an example:

```
R> est=debiasedMix$CausalEstimates
R> se=sqrt(diag(debiasedMix$VCovCausalEstimates))
R> Pmix=cbind(c("intercept",exposure_names), 2*pnorm(-abs(est)/se))
R> colnames(Pmix)=c("Causal Estimate","P-value")
```

5.2. Assessing pleiotropy and model fit

As stated earlier, it is a central assumption of MR that associations between the SNPs used as IVs and the outcome go only through the exposures included in MR. If there is evidence that an IV is associated with the outcome via alternative paths, causal estimates made using certain methods may be biased. The functions `MRBEE.IMRP()`, `MRBEE.Mix()`, and `MRBEE.IPOD()` all explicitly remove IVs with evidence of nonzero horizontal pleiotropy; `MRBEE.IMRP()` removes them using the IMRP (Zhu *et al.* 2021) procedure, `MRBEE.Mix()` uses a mixture distribution

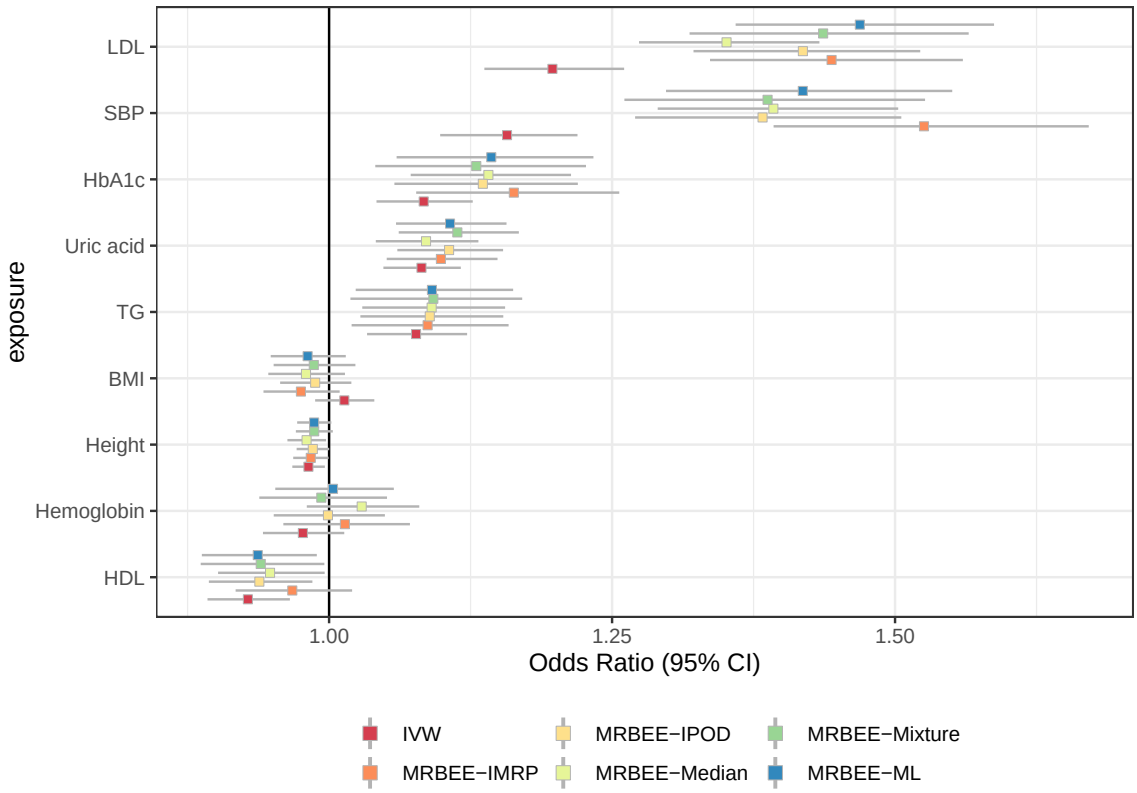

Figure 3: These are estimates for the causal effects of the listed exposures on coronary artery disease (CAD) risk in an East Asian population, the data for which are provided in the `cardioData` object. Estimates made by standard IVW are colored red and those made by bias-corrected robust MR methods using the MRBEE framework are presented in other colors.

to distinguish valid from horizontally pleiotropic IVs, and `MRBEE.IPOD()` uses the robust method of She and Owen (2011) to identify outliers. `MRBEE.Median()` assumes $< 50\%$ of IVs are horizontally pleiotropic and `MRBEE.MLqe()` assigns weights to IVs based on evidence of horizontal pleiotropy. We have created the function `residualPlot()` to visualize the ways in which these methods and their corresponding functions assesses evidence of horizontal pleitropy for each IV considered in MR. We demonstrate how this function is used below and show the results it can produce in Figure 4.

```
R> p1=residualPlot(pd,debiasedIMRP)+labs(title="Debiased IMRP")
R> p2=residualPlot(pd,debiasedIPOD)+labs(title="Debiased IPOD")
R> p3=residualPlot(pd,debiasedMix)+labs(title="Debiased Mixture")
R> p4=residualPlot(pd,debiasedML)+labs(title="Debiased ML")
R> ggpubr::ggarrange(p1,p2,p3,p4,nrow=2,ncol=2)
```

We have omitted a plot of horizontal pleiotropy for `MRBEE.Median()` because it technically neither weights nor removes specific IVs with evidence of horizontal pleiotropy. Figure 4 demonstrates that each method applies small (`MRBEE.ML()`) or zero (`MRBEE.IMRP()`, `MRBEE.Mix()`, `MRBEE.IPOD`) weights to IVs with evidence of horizontal pleiotropy. These IVs have the largest

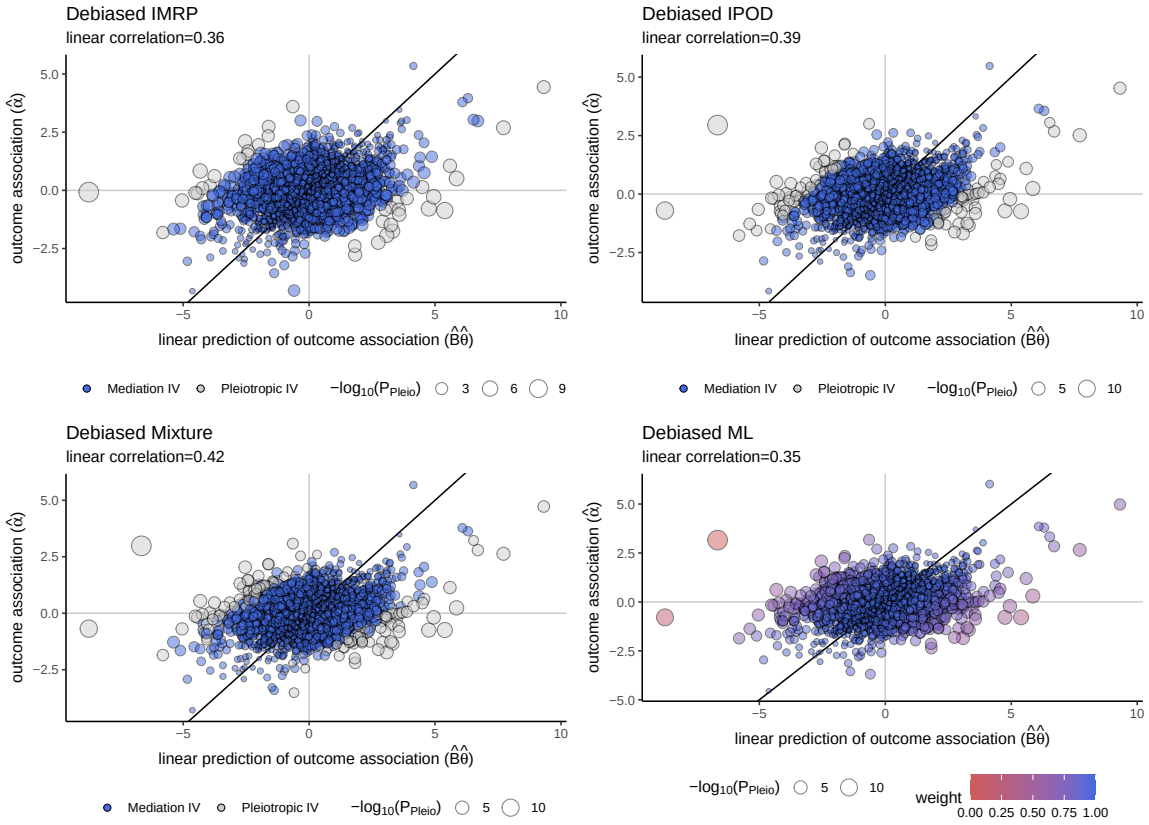

Figure 4: Each plot is produced using the `residualPlot()` function which takes as input `pd` (i.e., the output of `prepData()`) and a fitted model (i.e., the output of `MRBEE.IMRP()`, `MRBEE.Median()`, `MRBEE.Mix()`, `MRBEE.IPOD()`, and/or `MRBEE.MLqe()`). Results from `MRBEE.Median()` are omitted because it does not remove SNPs or explicitly downweight their associations due to horizontal pleiotropy. For Debiased IMRP, Debiased Mixture, and Debiased **Θ**-IPOD, grey points are considered horizontally pleiotropic and are not considered in causal estimation. Blue points are considered non-horizontally pleiotropic and are used for causal estimation. For Debiased ML, no points are removed from MR, but redder points received lower weight than bluer points. 'Linear correlation' values represent Pearson's r correlation between values on the y- and x-axes. For the Debiased IMRP, Debiased **Θ**-IPOD, and Debiased Mixture methods, these correlations are calculated use only the mediation (i.e., not horizontally pleiotropic IVs). For Debiased ML, the correlation is weighted by the IV weights using the `cov.wt()` function in the `stats` package

.

residuals in model fitting as displayed in Figure 4. We can observe from Figure 4 that the MRBEE-IMRP, MRBEE-IPOD, and MRBEE-Mix methods may exclude different subsets of the full IV set during causal estimation based on evidence of horizontal pleiotropy. Visual inspection of these results using `residualPlot()` can inform inference about which of these methods has most effectively addressed horizontal pleiotropy in MR.

P-values displayed in Figure 4 are from the test of horizontal pleiotropy in (16). The function `Spleio()` calculates these P-values and their corresponding test statistics. `Spleio()` receives as input `pd` (i.e., the output of `prepData()`) and two elements from a fitted model object. As an example, consider the fitted model object `debiasedMix` produced from the `MRBEE.Mix()` function. To calculate horizontal pleiotropy P-values based on the causal estimates stored in `debiasedMix`, we use the following code:

```
R> sp=Spleio(pd,debiasedMix)
R> names(sp)
[1] "Stats" "Ps"
R> head(cbind(sp$Stats,sp$Ps),1) # result for first IV
[1] 0.02597983 0.8719496
```

where the first element in the final output (0.02597983) is a chi-square statistic which follows a central chi-square distribution with degree of freedom 1 under the null hypothesis of no horizontal pleiotropy for the IV. These statistics and P-values can be used to calculate the commonly reported λ_{GC} genomic inflation factor (see Devlin and Roeder (1999)) for each debiased method. Consider the model from `MRBEE.Median()` as an example. First we must set a post-hoc P-value threshold to indicate a horizontally pleiotropic IV. For this example, we will set this threshold to 0.05. The code to calculate λ_{GC} for the non-horizontally pleiotropic IV set is

```
R> (lambdaGC=median(sp$Stats[sp$Ps>0.05])/qchisq(0.5,1))
[1] 1.10
```

When compared to the disjoint set of IVs (i.e., .those with `sp$Ps<0.05`, where $\lambda_{GC} = 12.78$), this is evidence that the `MRBEE.Median()` method performed MR with little influence from horizontally pleiotropic IVs.

5.3. Genome-wide horizontal pleiotropy testing

The test for horizontal pleiotropy can also be applied genome-wide to all SNPs in `cardioData` using the `genomePleio()` function. Inferences from genome-wide horizontal pleiotropy testing are still with respect to the MR exposures and outcome, considered in our examples to be CAD as the outcome and the 9 cardiometabolic conditions in Figure 3 as exposures. Evidence from applying this test genome-wide can identify novel loci and inform researchers about biological and epidemiological processes explaining observed genetic associations and pleiotropy between phenotypes (Lorincz-Comi *et al.* 2023; Zhu *et al.* 2022). For example, a novel locus significant in genome-wide horizontal pleiotropy testing but not standard GWAS may have a genetic correlation with one of the exposures used in the horizontal pleiotropy testing that is opposite the direction of its causal effect. Alternatively, genetic associations with the outcome that are completely mediated by the exposures will have very little association evidence in genome-wide horizontal pleiotropy testing. Interested readers are referred to Lorincz-Comi *et al.*

(2023); Zhu *et al.* (2022) for additional examples of real-world applications of this testing.

Earlier, we used the `biasTerms()` function on the full `cardioData` object to harmonise effect alleles between GWAS and calculate the correlation matrix $\mathbf{R}_{\mathbf{w}_\alpha \times \mathbf{W}_\beta}$ (equivalent to $\mathbf{\Sigma}_{W_\beta \times w_\alpha}$ when using Z-score standardization as we have) from (11) that is used to correct weak instrument bias. The output of this function can be used here to perform genome-wide testing. The `genomePleio()` function only requires specifying the output of `biasTerms()` and a user-chosen model fitted object from one of the following: `MRBEE.IMRP()`, `MRBEE.Median()`, `MRBEE.IPOD()`, `MRBEE.Mix()`, or `MRBEE.MLqe()`. Since genome-wide testing may be performed in an **R** session which may have limited computational resources, the function `genomePleio()` divides this large problem into `n_divisions` smaller problems. This function does not induce parallel computation, but users are free to perform these operations in their own parallel way using e.g. the `parallel` package in `R` wrapped around `genomePleio()`. Execution of `genomePleio()` is the following, where we use causal estimates from the output of `MRBEE.IPOD()`:

```
R> pD0=prepData(bT,oi=oi) # oi=1 defined above as the outcome index position
R> gwSp=genomePleio(pD0,debiasedIPOD,n_divisions=10)
R> names(gwSp)
[1] "PleioStat" "PleioP" "JointExpStat" "JointExpP" "OutcomeStat" "OutcomeP"
+   "aHat" "aHatHat"
```

The output of `genomePleio()` provides three main results: (i) horiozontal pleiotropy test statistics (`gwSp$PleioStat`) and their P-values (`gwSp$PleioP`), (ii) p-degree of freedom chi-square joint test statistics (`gwSp$JointExpStat`) for genetic associations with the p exposures and their P-values (`gwSp$JointExpP`), (iii) chi-square statistics for genetic associations with the outcome (`sqSp$OutcomeStat`) and their P-values (`gwSp$OutcomeP`). For the jth SNP tested, the chi-square joint test for the p exposures tests $H_{0j} : \beta_{1j} = ... = \beta_{pj} = 0$ where β_{kj} is the true association between the jth SNP and kth exposure. Two additional results (`gwSp$aHat`, and `gwSp$aHatHat`) are provided which respectively represent estimated effect sizes of the SNPs on the outcome ($\hat{\boldsymbol{\alpha}}$) and the linear predictor of these associations given the exposure associations and estimated causal effects ($\widehat{\mathbf{B}}\hat{\boldsymbol{\theta}}$). These are not test statistics but may be useful later when comparing effect size directions between phenotypes. Computation using the command above is performed for sequential chunks of approximate size M/`n_divisions` where M is the total number of available SNPs genome-wide. This process takes approximately 18 minutes to complete on a desktop computer with a 3.20Ghz processor and 32Gb of RAM when using 9 exposures and 6.7M SNPs. High-performance computers could accomplish this task in much less time.

We can use the `gwPleioPlot()` function to display the results of genome-wide horizontal pleiotropy testing. Note that if the user is only interested in the results of this testing, and not also in comparing the results of this testing to the original GWAS results for the outcome phenotype, any standard package for creating Manhattan plots can be used (e.g., the **CMplot** package in `R`: https://github.com/YinLiLin/CMplot). First, we can calculate the genomic inflation value λ_{GC} for the horizontal pleiotropy tests and joint exposure tests using the following commands:

```
R> (median(gwSp$Stats,na.rm=T)/qchisq(0.5,1))
[1] 1.116908 # lambdaGC for genome-wide horizontal pleiotropy test
```

```
R> (median(gwSp$JointExpStat,na.rm=T)/qchisq(0.5,9))
[1] 1.193351 # lambdaGC for joint exposures GWAS
R> (median(gwSp$OutcomeStat,na.rm=T)/qchisq(0.5,1))
[1] 1.156405 # lambdaGC for outcome GWAS
```

The `gwPleioPlot()` function displays genome-wide pleiotropy test results and compares them to original GWAS results for the outcome phenotype, which in our examples is CAD. The function `gwPleioPlot()` creates multiple figures and places them into a single list to be used for visualization. These figures include a QQ plot, Manhattan plot, and a locus-specific plot of horizontal pleiotropy for a user-identified list of loci tagged by lead SNPs whose row indices in `data` are in `leadSNPs`. The output of `gwPleioPlot()` is a list with elements that are each of these plots. These will be `ggplot2` objects and so can be modified by the user after they have been created. Note that since genome-wide testing is often performed for millions of SNPs, displaying these figures for all SNPs can be very time consuming. The `GWsubInds` argument in `gwPleioPlot()` can be used to display only a subset of all SNPs genome-wide. The argument `GWsubInds` takes a vector of row-indices to use in creating whole-genome plots and defaults to the full set of SNPs. The argument `leadSNPs` is a vector of row indices in `cardioData` corresponding to lead SNPs whose associations with the outcome tag independent loci and must be specified to produce a locus-specific plot. The vector of row indices given to the `GWsubInds` argument will not affect locus specific plots from loci specified by lead SNPs in the `leadSNPs` argument. Since `gwPleioPlot()` will produce a Manhattan plot, the user must also provide a vector of CHR and BP positions in the format `<CHR>_<BP>` to the argument `CHR_BP` in `gwPleioPlot()`. Assuming the `CHR_BP` vector has already been created, we can visualize results of genome-wide pleiotropy testing for all SNPs using `gwPleioPlot()` with the following commands:

```
R> plots=gwPleioPlot(gwSp,leadSNPs,CHR_BP)
R> names(plots)
[1] "QQ" "Manhattan" "LocusPlot"
R> print(plots$QQ)
R> print(plots$Manhattan)
R> print(plots$LocusPlot)
R> w=c(4,10,5) # setting widths of images to save
R> for(i in 1:3) {  # save plots
+   ggsave(paste0(names(plots)[i], ".png"),height=5,width=w[i],units="cm")
+ }
```

The three outputs of `gwPleioPlot()` are displayed in Figures 5, 6, and 7. These results are helpful in better understanding how certain loci may be associated with the outcome trait (CAD in our examples) through a set of exposures (9 cardiometabolic risk factors for CAD in our examples; see Figure 3).

6. Commmand-line MRBEE

In this section, we demonstrate how to use the MRBEE command line tool to estimate causal effects between a set of risk factors and an outcome. As mentioned previously, MRBEE can also be performed using the **Python** language instead of **R** in one of two formats. Researchers

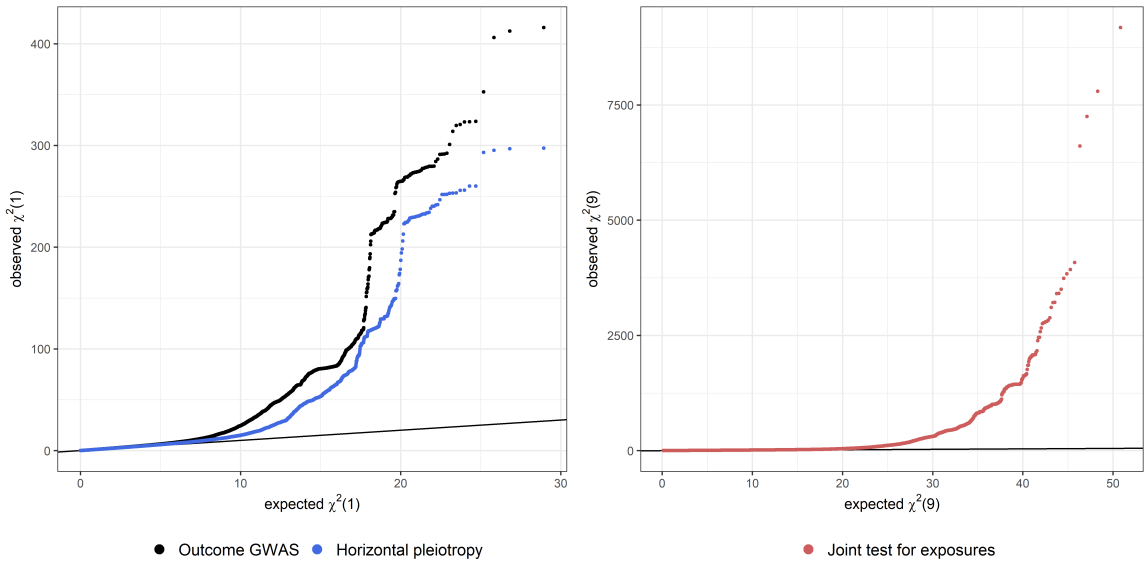

Figure 5: QQ-plot for the outcome GWAS (left), a joint test for 9 cardiometabolic exposures described in the text (right), and genome-wide horizontal pleiotropy testing (left) using causal estimates from the MRBEE-IPOD method using the `MRBEE.IPOD()` function. Under the null hypothesis, a horizontal pleiotropy test statistic for 1 outcome will follow a central chi-square distribution with 1 degree of freedom and a joint test statistic for p exposures will follow a central chi-square distribution with p degrees of freedom (right).

may either directly source the full set of commands from `mrbeesrc.py` at the github repository `https://github.com/noahlorinczcomi/MRBEE`, or they may use the command-line tool `mrbeecmd.py` stored at the same repository. For brevity in this paper, a complete demonstration of how to use the **MRBEE Python** module from `mrbeesrc.py` is reserved for the online tutorial located at `https://github.com/noahlorinczcomi/MRBEE/PyTutorial.html` and we only introduce the command line tool `mrbeecmd.py` here. For each exposure and the outcome, the MRBEE command line tool requires only GWAS effect sizes, standard errors, and a unique SNP identifier. As before with the MRBEE **R** package, this **Python** tool assumes that a list of IVs to use in MR, named by their unique SNP identifiers, is already available in the file `IVs.txt`. Currently, both the command line tool and the **Python** module only implement the MRBEE-IMRP method.

The user is required to specify 4 column names for GWAS summary statistics from each exposure and the outcome that provide column names for the following information: (i) unique SNP identifier (of consistent format across all phenotypes), (ii) association/effect size estimate, (iii) standard error of association/effect size estimate, (iv) effect allele. This tool performs allele harmonization and by default uses the Z-score standardization (Lorincz-Comi *et al.* 2023; Zhu *et al.* 2015) for all association estimates and their standard errors. If researchers wish to use a different standardization, the GWAS association estimates and their standard errors should already be in their preferred standardized scale when given to the MRBEE command line tool and it should be indicated that the additional Z-score standardization should not be performed. We begin by showing a full example of how to use this tool for CAD (CARDIoGRAMplusC4D 2015) and exposures HbA1c and uric acid

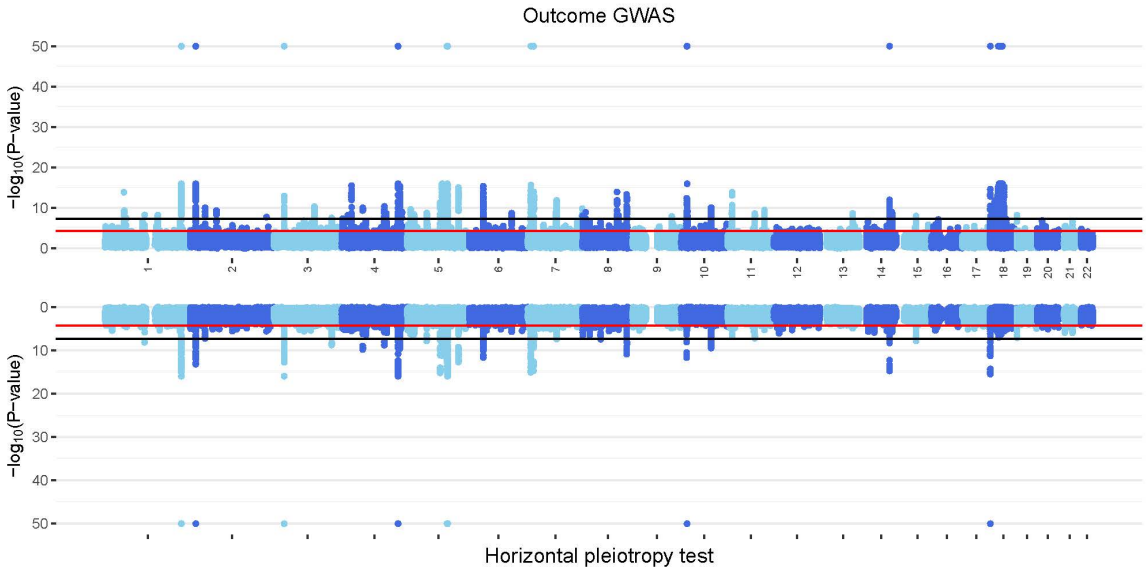

Figure 6: Manhattan plots for the outcome GWAS (top) and genome-wide horizontal pleiotropy testing (bottom) using causal estimates from the MRBEE-IPOD method using the `MRBEE.IPOD()` function. By default, this plot is restricted only to SNPs with $P < 0.05$ in the outcome GWAS or genome-wide horizontal pleiotropy testing. This threshold can be changed (e.g., to 1 to show all SNPs) using the `manhattanPUpper` command of the `gwPleioPlot()` function. Loci detected in the outcome GWAS but not genome-wide horizontal pleiotropy testing may have associations with the outcome that are mediated by the MR exposures.

(http://www.nealelab.is/uk-biobank), then describe each of its components in greater detail. First, we assume that the `mrbeecmdfunctions.py` and `mrbeecmd.py` files have been download from https://github.com/noahlorinczcomi/MRBEE and are located in the current working directory. MR and genome-wide horizontal pleiotropy testing using the MRBEE-IMRP method can be performed using the following command line expression:

```
~$ mrbeecmd.py \
~$ -exposureGWAS hba1c_gwas.txt.gz,uricacid_gwas.txt.gz \
~$ -outcome GWAS cad_gwas.txt.gz \
~$ -IVs IVs.txt \
~$ -exposureSNP rsid,rsid \
~$ -outcomeSNP markername \
~$ -exposureBETA beta,beta \
~$ -exposureSE se,se \
~$ -outcomeBETA beta \
~$ -outcomeSE se_dgc \
~$ -outcomeEffectAllele effect_allele \
~$ -exposureEffectAllele alt,alt \
~$ -stdZ true \
~$ -genomewideSpleio true \
~$ -out results \
```

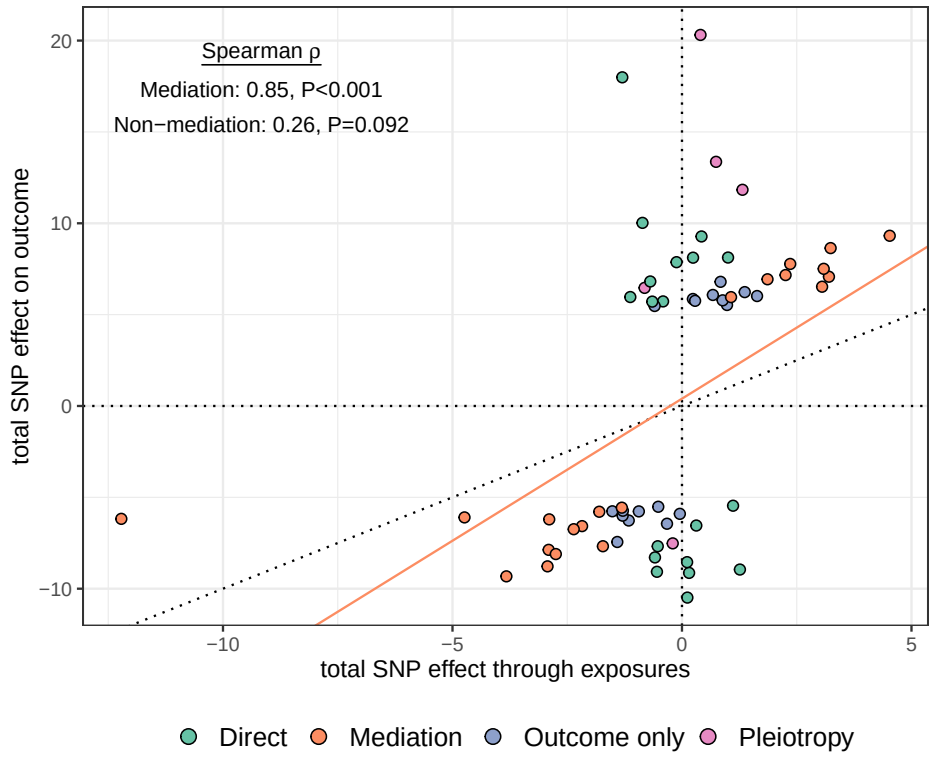

Figure 7: The output of `gwPleioPlot(<x>)$LocusPlot` using causal estimates from MRBEE-IPOD method using the `MRBEE.IPOD()` function. The user specifies a vector of row indices in the original full GWAS data (`cardioData`) corresponding to lead SNPs tagging loci associated with the outcome trait and the `classifyLoci()` function called by the `gwPleioPlot()` classifies them as either mediation, pleiotropy, direct, outcome only, or novel pleiotropy loci. These determinations are made based on a user-specified level of genome-wide significance ($P < 5 \times 10^{-8}$ by default) and following the algorithm in Figure 4b of Lorincz-Comi *et al.* (2023). 'Mediation' loci are associated with the outcome in GWAS but not horizontal pleiotropy testing; 'pleiotropy' loci are associated in both tests and in a joint test for assocation with any of the exposures; 'direct' loci are associated with the outcome in the outcome GWAS and horizontal pleiotropy testing but not in the joint test for the exposures; 'outcome only' loci are associated with the outcome only in the outcome GWAS; 'novel pleiotropy' loci are associated with the outcome in horizontal pleiotropy testing but not the outcome GWAS. Generally, the distance of a lead SNP from the diagonal line (dotted black) is inversely proportional to the degree to which its association with the outcome is mediated by the exposures. For example, a lead SNP on the diagonal line has an association with the outcome that is completely mediated by the exposures.

A description of the flags used in this command are available in Table 6. Running the above commands prints the following output to the console:

```
Getting things ready
Reading in outcome GWAS data
Reading in GWAS data for exposure 1
Reading in GWAS data for exposure 2
Harmonizing alleles
Calculating bias-correction terms
Performing MR
Using 1445 candidate SNPs in MR
31 SNPs removed from MR because of horizontal pleiotropy, 1414 SNPs remain
Performing genome-wide horizontal pleiotropy testing using Spleio
10% of genome-wide Spleio testing complete
...
100% of genome-wide Spleio testing complete
Saving results to results_<causal_estimates,MR_data,genomewide_Spleio>.csv
```

For the example presented above, the program took 2.25 minutes to complete. Currently, the command line program `mrbeecmd.py` is equivalent to the `MRBEE.IMRP()` method implemented in **R** that was introduced above. Future work will allow users to choose between one or more of the MRBEE-IMRP, MRBEE-IPOD, MRBEE-Median, MRBEE-MLqe, and MRBEE-Mixture methods when using this tool.

7. Summary

We developed the **MRBEE R** package, **Python** module, and command line tool for performing causal estimation and genome-wide horizontal pleiotropy testing. These software are the first to allow users to perform robust MVMR while correcting for biases from weak instruments, measurement error, and sample overlap. We introduced new statistical methods to correct these biases while remaining robust to horizontal pleiotropy, one of the most frequently encountered challenges in MR applications. We additionally provide functionality for users to rigorously investigate the extent of horizontal pleiotropy in MR. Our package is also the first that can be used for genome-wide horizontal pleiotropy testing, which has been successfully used for identifying novel genetic associations and their potential causal pathways to disease outcomes in coronary artery disease and hypertension (Lorincz-Comi *et al.* 2023; Zhu *et al.* 2022).

The functions in the **R** package **MRBEE** are divided into three groups: (i) *Preparing Data*, (ii) *Performing MR*, (iii) *Pleiotropy Analysis*. These groups represent the ordered steps required to perform a comprehensive MR and pleiotropy analysis. To make use of **MRBEE** intuitive and simple, the direct output of commands executed in earlier steps are used as input in later steps. As a result, the user is only required to specify minimal input to receive their main results. All functions are also computationally fast for the magnitude of operations they perform. In our examples with 6.7M SNPs genome-wide, 3,097 IVs used in MR, and 9 MR exposures, `bT()` completes in ∼48secs, `prepData()` in ∼2mins, `Spleio()` on the IVs in <1secs, and `genomePleio()` in ∼18mins. Computation times using the class of `MRBEE.<x>()`

Flag	Description	Default
`-exposureGWAS`	Comma-separated list of full .gz filepaths to exposure GWAS data sets[a]	-
`-outcome`	Full .gz filepath to outcome GWAS data set[a]	-
`-IVs`	Filepath to file containing unique SNP identifiers for candidate IVs[b]	-
`-exposureSNP`	Comma-separated list of unique SNP identifier column names in exposure GWAS	`SNP,...,SNP`
`-outcomeSNP`	Unique SNP identifier column name in outcome GWAS	`SNP`
`-exposureBETA`	Comma-separated list of unique effect size column names in exposure GWAS[c]	`BETA,...,BETA`
`-exposureSE`	Comma-separated list of unique standard error column names in exposure GWAS	`SE,...,SE`
`-outcomeBETA`	Effect size column name in outcome GWAS	`BETA`
`-outcomeSE`	Standard error column name in outcome GWAS	`SE`
`-exposureEffectAllele`	Comma-separated list of effect allele column names in exposure GWAS	`ALT`
`-outcomeEffectAllele`	Effect allele column name in outcome GWAS	`ALT,...,ALT`
`-stdZ`	Should MR be performed using BETA/SE for all exposures and outcome? (True, False)[d]	`TRUE`
`-genomewideSpleio`	Should genome-wide Spleio testing be performed? (True, False)	`FALSE`
`-out`	Name of output files[e]	`results`

Table 1: The order of elements in comma-separated lists of names for multiple exposures are assumed to be consistent for the lists of SNP identifier column names, effect size column names, standard error column names, and effect allele column names. [a]: Files must have the .gz extension. [b]: Can have any common extension (e.g., .txt, .tsv, .csv, .tab) or delimiter (e.g., ',', ' ', tab), but must contain only a single column and the unique SNP identifiers must have a format that appears in the unique SNP identifiers specified for the exposures and outcome. Most simply, this will be rsIDs. [c]: 'effect sizes' means standardized association estimates or regression coefficients. [d] By default, GWAS standardized effect sizes will be converted to their corresponding Z-scores for use in MR. If you do not wish to make this or any additional standardization, include `-stdZ false`. [e]: 2 .csv files are written out by default: (1) table of causal estimates, standard errors, and corresponding P-values, (2) table of all data used in MR and indications of horizontal pleiotropy for each IV.

functions on these data are similarly fast, ranging in computation time from <0.1secs for `MRBEE.MLqe()` to ∼1.1mins for `MRBEE.Median()`. This is a major advantage over existing software (e.g., `R` packages `cause`, `MRPRESSO`, `MR.CUE`, and unpackaged `MR-BMA` at https://github.com/verena-zuber/demo_AMD) for performing MR that can take upwards of 10 minutes just to estimate the causal effect of a single exposure on an outcome using a relatively small number of IVs (e.g., <250). We provide a computationally efficient **R** package (**MRBEE**) to perform multivariable Mendelian Randomization using GWAS summary statistics that corrects for bias from weak instruments, sample overlap, measurement error, and horizontal pleiotropy.

References

Bowden J, *et al.* (2015). "Mendelian randomization with invalid instruments: effect estimation and bias detection through Egger regression." *International Journal of Epidemiology*, **44**(2), 512–525.

Bowden J, *et al.* (2016). "Consistent estimation in Mendelian randomization with some invalid instruments using a weighted median estimator." *Genetic Epidemiology*, **40**(4), 304–314.

Burgess S, Thompson SG (2015). "Multivariable Mendelian randomization: the use of pleiotropic genetic variants to estimate causal effects." *American Journal of Epidemiology*, **181**(4), 251–260.

Burgess S, *et al.* (2011). "Avoiding bias from weak instruments in Mendelian randomization studies." *International Journal of Epidemiology*, **40**(3), 755–764.

Burgess S, *et al.* (2013). "Mendelian randomization analysis with multiple genetic variants using summarized data." *Genetic Epidemiology*, **37**(7), 658–665.

Burgess S, *et al.* (2023). "Using genetic association data to guide drug discovery and development: Review of methods and applications." *The American Journal of Human Genetics*, **110**(2), 195–214.

CARDIoGRAMplusC4D (2015). "A comprehensive 1000 Genomes–based genome-wide association meta-analysis of coronary artery disease." *Nature Genetics*, **47**(10), 1121–1130.

Cheng Q, *et al.* (2022). "Mendelian randomization accounting for complex correlated horizontal pleiotropy while elucidating shared genetic etiology." *Nature Communications*, **13**(1), 1–13.

Corbin LJ, *et al.* (2016). "BMI as a modifiable risk factor for type 2 diabetes: refining and understanding causal estimates using Mendelian randomization." *Diabetes*, **65**(10), 3002–3007.

Davey Smith G, Ebrahim S (2003). "'Mendelian randomization': can genetic epidemiology contribute to understanding environmental determinants of disease?" *International journal of epidemiology*, **32**(1), 1–22.

Devlin B, Roeder K (1999). "Genomic control for association studies." *Biometrics*, **55**(4), 997–1004.

Fan J, Li R (2001). "Variable selection via nonconcave penalized likelihood and its oracle properties." *Journal of the American statistical Association*, **96**(456), 1348–1360.

Ferrari D, Yang Y (2010). "Maximum Lq-likelihood estimation." *The Annals of Statistics*, **38**(2), 753–783.

Grant AJ, Burgess S (2021). "Pleiotropy robust methods for multivariable Mendelian randomization." *Statistics in Medicine*, **40**(26), 5813–5830.

Hemani G, *et al.* (2020). ***TwoSampleMR**: Mendelian randomization with GWAS summary data*. R package version 0.8-1, URL https://mrcieu.github.io/TwoSampleMR/.

Hu X, *et al.* (2022). "Mendelian randomization for causal inference accounting for pleiotropy and sample structure using genome-wide summary statistics." *Proceedings of the National Academy of Sciences*, **119**(28), e2106858119.

Huber PJ (2011). "Robust statistics." In *International encyclopedia of statistical science*, pp. 1248–1251. Springer.

Kang H, *et al.* (2016). "Instrumental variables estimation with some invalid instruments and its application to Mendelian randomization." *Journal of the American Statistical Association*, **111**(513), 132–144.

Liu DC, Nocedal J (1989). "On the limited memory BFGS method for large scale optimization." *Mathematical Programming*, **45**(1-3), 503–528.

Lorincz-Comi N, *et al.* (2023). "MRBEE: A novel bias-corrected multivariable Mendelian randomization method." *biorxiv, 523480*.

MacArthur J, *et al.* (2017). "The new NHGRI-EBI Catalog of published genome-wide association studies (GWAS Catalog)." *Nucleic Acids Research*, **45**(D1), D896–D901.

Mailman MD, *et al.* (2007). "The NCBI dbGaP database of genotypes and phenotypes." *Nature Genetics*, **39**(10), 1181–1186.

Morrison J (2019). ***CAUSE**: a Mendelian Randomization method using genome-wide summary statistics*. R package, URL https://jean997.github.io/cause/.

Morrison J, *et al.* (2020). "Mendelian randomization accounting for correlated and uncorrelated pleiotropic effects using genome-wide summary statistics." *Nature Genetics*, **52**(7), 740–747.

Purcell S, *et al.* (2007). "PLINK: a tool set for whole-genome association and population-based linkage analyses." *The American journal of human genetics*, **81**(3), 559–575.

Qi G, Chatterjee N (2019). "Mendelian randomization analysis using mixture models for robust and efficient estimation of causal effects." *Nature Communications*, **10**(1), 1–10.

Qing C, *et al.* (2022). ***MR.CUE**: a package for Mendelian randomization informs shared genetic etiology underlying exposure and outcome by interrogating correlated horizontal pleiotropy*. R package, URL https://github.com/QingCheng0218/MR.CUE.

Qing G, Chatterjee N (2019). ***MRMix**: R package for two-sample Mendelian randomization*. R package, URL https://github.com/gqi/MRMix.

Rees JM, *et al.* (2017). "Extending the MR-Egger method for multivariable Mendelian randomization to correct for both measured and unmeasured pleiotropy." *Statistics in Medicine*, **36**(29), 4705–4718.

Rees JM, *et al.* (2019). "Robust methods in Mendelian randomization via penalization of heterogeneous causal estimates." *PloS One*, **14**(9), e0222362.

Sadreev II, *et al.* (2021). "Navigating sample overlap, winner's curse and weak instrument bias in Mendelian randomization studies using the UK Biobank." *medRxiv*.

Sanderson E, *et al.* (2019). "An examination of multivariable Mendelian randomization in the single-sample and two-sample summary data settings." *International Journal of Epidemiology*, **48**(3), 713–727.

Sanderson E, *et al.* (2021). "Testing and correcting for weak and pleiotropic instruments in two-sample multivariable Mendelian randomization." *Statistics in Medicine*, **40**(25), 5434–5452.

Sanderson E, *et al.* (2022). "Mendelian randomization." *Nature Reviews Methods Primers*, **2**(1), 1–21.

She Y, Owen AB (2011). "Outlier detection using nonconvex penalized regression." *Journal of the American Statistical Association*, **106**(494), 626–639.

Shen J, *et al.* (2021). "A modifiable risk factors atlas of lung cancer: A Mendelian randomization study." *Cancer Medicine*, **10**(13), 4587–4603.

Stefanski LA, Cook JR (1995). "Simulation-Extrapolation: The Measurement Error Jackknife." *Journal of the American Statistical Association*, **90**(432), 1247–1256. ISSN 01621459. URL http://www.jstor.org/stable/2291515.

Tibshirani R (1996). "Regression shrinkage and selection via the lasso." *Journal of the Royal Statistical Society: Series B (Methodological)*, **58**(1), 267–288.

Vaucher J, *et al.* (2018). "Cannabis use and risk of schizophrenia: a Mendelian randomization study." *Molecular Psychiatry*, **23**(5), 1287–1292.

Verbanck M, *et al.* (2018). "Detection of widespread horizontal pleiotropy in causal relationships inferred from Mendelian randomization between complex traits and diseases." *Nature Genetics*, **50**(5), 693–698.

Wang H, *et al.* (2012). "Corrected-loss estimation for quantile regression with covariate measurement errors." *Biometrika*, **99**(2), 405–421.

Wang K, *et al.* (2022). "Mendelian randomization analysis of 37 clinical factors and coronary artery disease in East Asian and European populations." *Genome Medicine*, **14**(1), 1–15.

Yavorska OO, Burgess S (2017). "MendelianRandomization: an R package for performing Mendelian randomization analyses using summarized data." *International Journal of Epidemiology*, **46**(6), 1734–1739.

Ye T, *et al.* (2021). "Debiased inverse-variance weighted estimator in two-sample summary-data Mendelian randomization." *The Annals of Statistics*, **49**(4), 2079–2100.

Yi GY (2017). *Statistical analysis with measurement error or misclassification: strategy, method and application*. Springer.

Zhao Q, *et al.* (2020). "Statistical inference in two-sample summary-data Mendelian randomization using robust adjusted profile score." *The Annals of Statistics*, **48**(3), 1742–1769.

Zhu X (2020). "Mendelian randomization and pleiotropy analysis." *Quantitative Biology*, pp. 1–11.

Zhu X, *et al.* (2015). "Meta-analysis of correlated traits via summary statistics from GWASs with an application in hypertension." *The American Journal of Human Genetics*, **96**(1), 21–36.

Zhu X, *et al.* (2021). "An iterative approach to detect pleiotropy and perform Mendelian Randomization analysis using GWAS summary statistics." *Bioinformatics*, **37**(10), 1390–1400.

Zhu X, *et al.* (2022). "Genome-wide pleiotropy analysis identifies novel blood pressure variants and improves its polygenic risk scores." *Genetic Epidemiology*, **46**(2), 105–121.

Affiliation:

Xiaofeng Zhu, Ph.D
Professor
Department of Population and Quantitative Health Sciences, School of Medicine,
Case Western Reserve University
Wolstein Research Building
2103 Cornell Road
Cleveland, Ohio 44106
Phone: 216-368-0201
Fax: 216-368-4880
Email: xxz10@case.edu
URL: https://case.edu/medicine/pqhs/about/people/primary-faculty/xiaofeng-zhu

Journal of Statistical Software http://www.jstatsoft.org/
published by the Foundation for Open Access Statistics http://www.foastat.org/
MMMMMM YYYY, Volume VV, Issue II *Submitted:* yyyy-mm-dd
doi:10.18637/jss.v000.i00 *Accepted:* yyyy-mm-dd