

Fine-mapping from summary data with the “Sum of Single Effects” model—S1 Text

Yuxin Zou¹, Peter Carbonetto^{2,3}, Gao Wang^{4*}, Matthew Stephens^{1,2*}

1 Department of Statistics, University of Chicago, Chicago, Illinois, United States of America

2 Department of Human Genetics, University of Chicago, Chicago, Illinois, United States of America

3 Research Computing Center, University of Chicago, Chicago, Illinois, United States of America

4 Department of Neurology and the Gertrude. H. Sergievsky Center, Columbia University, New York, New York, United States of America

* wang.gao@columbia.edu

* mstephens@uchicago.edu

Detailed methods

IBSS-ss algorithm

SuSiE-RSS uses the IBSS-ss algorithm, which is essentially the IBSS algorithm [1] modified for sufficient statistics or summary statistics. When applied to sufficient statistics, the IBSS-ss algorithm returns the same results as the IBSS algorithm applied to the individual-level data. Here we describe the IBSS-ss algorithm in detail.

The single effect regression (SER) model with summary statistics

The single effect regression (SER) model is defined in [1] (see also [2]) as a multiple regression model in which exactly one variable has a non-zero effect on the outcome. It is a special case of the *SuSiE* model when $L = 1$. Posterior computations with the SER model form the basis for the *SuSiE* model fitting algorithm, IBSS, hence form the basis for IBSS-ss. We show here that posterior quantities under the SER model can be computed using summary statistics $\mathbf{X}^\top \mathbf{X}$ and $\mathbf{X}^\top \mathbf{y}$.

Formally, the SER model can be written as

$$\mathbf{y} = \mathbf{X}\mathbf{b} + \mathbf{e} \tag{28}$$

$$\mathbf{e} \sim \mathcal{N}_N(0, \sigma^2 \mathbf{I}_N) \tag{29}$$

$$\mathbf{b} = \gamma b \tag{30}$$

$$\gamma \sim \text{Multinomial}(1, \boldsymbol{\pi}) \tag{31}$$

$$b \sim \mathcal{N}(0, \sigma_0^2). \tag{32}$$

Here, $\mathbf{y} = (y_1, \dots, y_N)^\top \in \mathbb{R}^N$ denotes the phenotypes of N individuals, $\mathbf{X} \in \mathbb{R}^{N \times J}$ denotes their corresponding genotypes at J genetic variants (SNPs), $\mathbf{b} = (b_1, \dots, b_J)^\top$ denotes a vector of regression coefficients, \mathbf{e} is an N -vector of error terms, $\sigma^2 > 0$ is the residual variance parameter, and \mathbf{I}_N is the $N \times N$ identity matrix. To simplify the presentation, we assume \mathbf{y} and the columns of \mathbf{X} are centered to have mean zero, which avoids the need for an intercept term [3]. $\mathcal{N}(\mu, \sigma^2)$ denotes the univariate normal

distribution with mean $\boldsymbol{\mu}$ and variance $\boldsymbol{\Sigma}$, $\mathcal{N}_r(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes the r -variate normal distribution with mean $\boldsymbol{\mu}$ and variance $\boldsymbol{\Sigma}$, and $\text{Multinomial}(n, \mathbf{p})$ denotes the multinomial distribution with n trials and category probabilities $\mathbf{p} = (p_1, \dots, p_J)$. Thus, $\boldsymbol{\gamma} = (\gamma_1, \dots, \gamma_J) \in \{0, 1\}^J$ is a binary vector of length J in which exactly one element is 1 and the rest are 0, and so \mathbf{b} is a vector with exactly one non-zero element (except for the special case when $b = 0$). The scalar b gives the value of the one non-zero element in \mathbf{b} (the ‘‘single effect’’). The prior inclusion probabilities, $\boldsymbol{\pi} = (\pi_1, \dots, \pi_J)$, which we assume are fixed and known, give the prior probability that each genetic variant has the non-zero effect. The prior variance of the single effect, σ_0^2 , and the residual variance, σ^2 , are hyperparameters that can be pre-specified or, more commonly, estimated.

Given settings of the hyperparameters σ^2, σ_0^2 , the posterior distribution of \mathbf{b} under the SER model is worked out in [1]. We summarize it here, with a focus on performing computations using summary statistics $\mathbf{X}^\top \mathbf{X}$ and $\mathbf{X}^\top \mathbf{y}$.

Proposition 1. Consider the SER model (28–32) with known σ_0^2 and σ^2 . The posterior distribution of $\mathbf{b} = \boldsymbol{\gamma}b$ can be expressed in terms of univariate least-squares estimates of b_j , $\hat{b}_j := \mathbf{x}_j^\top \mathbf{y} / \mathbf{x}_j^\top \mathbf{x}_j$, and their variances, $s_j^2 := \sigma^2 / \mathbf{x}_j^\top \mathbf{x}_j$. Specifically, the posterior distribution of $\boldsymbol{\gamma}$ is

$$\boldsymbol{\gamma} \mid \mathbf{X}, \mathbf{y}, \sigma^2, \sigma_0^2 \sim \text{Multinomial}(1, \boldsymbol{\alpha}) \quad (33)$$

and the posterior distribution of b given $\boldsymbol{\gamma}$ is

$$b \mid \mathbf{X}, \mathbf{y}, \sigma^2, \sigma_0^2, \boldsymbol{\gamma}_j = 1 \sim \mathcal{N}(\mu_{1j}, \sigma_{1j}^2), \quad (34)$$

where

$$\sigma_{1j}^2 := \frac{1}{1/\sigma_0^2 + 1/s_j^2} \quad (35)$$

$$\mu_{1j} := \sigma_{1j}^2 \hat{b}_j / s_j^2 \quad (36)$$

$$\alpha_j := \frac{\pi_j \text{BF}_j}{\sum_{j'=1}^J \pi_{j'} \text{BF}_{j'}} \quad (37)$$

$$\begin{aligned} \text{BF}_j &:= \text{BF}(\mathbf{x}_j, \mathbf{y}; \sigma^2, \sigma_0^2) \\ &:= \frac{p(\mathbf{y} \mid \mathbf{x}_j, \sigma^2, \sigma_0^2)}{p(\mathbf{y} \mid \mathbf{x}_j; \sigma^2, b = 0)} \\ &= \sqrt{\frac{s_j^2}{\sigma_0^2 + s_j^2}} \times \exp\left(\frac{\hat{b}_j^2}{2s_j^2} \times \frac{\sigma_0^2}{\sigma_0^2 + s_j^2}\right). \end{aligned} \quad (38)$$

Note that the α_j ’s are the PIPs (2) under the SER model; $\text{PIP}_j = \alpha_j$, $j = 1, \dots, J$.

Proposition 1 shows that, given σ^2 and σ_0^2 , the posterior for \mathbf{b} under the SER model can be computed from the least-squares estimates \hat{b}_j and variances s_j^2 . These, in turn, can be computed from $\mathbf{X}^\top \mathbf{X}$ and $\mathbf{X}^\top \mathbf{y}$. We define SER-ss as the function that returns the posterior distribution of \mathbf{b} under the SER model given these statistics:

$$\text{SER-ss}(\mathbf{X}^\top \mathbf{X}, \mathbf{X}^\top \mathbf{y}; \sigma^2, \sigma_0^2) := (\boldsymbol{\alpha}, \boldsymbol{\mu}_1, \boldsymbol{\sigma}_1^2), \quad (39)$$

where $\boldsymbol{\mu}_1 := (\mu_{11}, \dots, \mu_{1J})^\top$, $\boldsymbol{\sigma}_1^2 := (\sigma_{11}^2, \dots, \sigma_{1J}^2)$ and $\boldsymbol{\alpha} := (\alpha_1, \dots, \alpha_J)$.

Remark 1. Although we write the posterior SER-ss as a function of $\mathbf{X}^\top \mathbf{X}$, it actually only depends on the diagonal elements of this matrix.

Algorithm 1 Iterative Bayesian stepwise selection using sufficient or summary statistics (IBSS-ss)

Require: Sufficient statistics $\mathbf{X}^\top \mathbf{X}$, $\mathbf{X}^\top \mathbf{y}$ and, optionally, $\mathbf{y}^\top \mathbf{y}$, N (for estimating σ^2).

Alternatively, require summary data that can be used to recover these sufficient statistics, either exactly or approximately. For example, if \mathbf{X} , \mathbf{y} are standardized, $\mathbf{X}^\top \mathbf{X}$, $\mathbf{X}^\top \mathbf{y}$ can be recovered from $\bar{\mathbf{R}}$, $\hat{\mathbf{z}}$, N .

Require: Number of effects, L ; initial estimates of hyperparameters σ^2 , σ_0^2 .

Require: Initial estimates of the posterior mean single effects, $\bar{\mathbf{b}}_l$, for $l = 1, \dots, L$.

```

1: repeat
2:    $\bar{\boldsymbol{\rho}} \leftarrow \mathbf{X}^\top \mathbf{y} - \mathbf{X}^\top \mathbf{X} \sum_{l=1}^L \bar{\mathbf{b}}_l$  ▷ Compute expected residuals.
3:   for  $l$  in  $1, \dots, L$  do
4:      $\bar{\boldsymbol{\rho}}_l \leftarrow \bar{\boldsymbol{\rho}} + \mathbf{X}^\top \mathbf{X} \bar{\mathbf{b}}_l$  ▷ Disregard  $l$ th single effect in residuals.
5:      $\sigma_{0l}^2 \leftarrow \operatorname{argmax}_{\sigma_0^2} \ell_{\text{SER-ss}}(\sigma^2, \sigma_0^2; \mathbf{X}^\top \mathbf{X}, \bar{\boldsymbol{\rho}})$  ▷ Update  $\sigma_{0l}^2$  (optional).
6:      $(\boldsymbol{\alpha}_l, \boldsymbol{\mu}_{1l}, \sigma_{1l}^2) \leftarrow \text{SER-ss}(\mathbf{X}^\top \mathbf{X}, \bar{\boldsymbol{\rho}}_l; \sigma^2, \sigma_{0l}^2)$  ▷ Fit SER to residuals.
7:      $\bar{\mathbf{b}}_l \leftarrow \boldsymbol{\alpha}_l \circ \boldsymbol{\mu}_{1l}$  ▷ “ $\circ$ ” denotes element-wise multiplication.
8:      $\bar{\mathbf{b}}_l^2 \leftarrow \boldsymbol{\alpha}_l \circ (\boldsymbol{\mu}_{1l} \circ \boldsymbol{\mu}_{1l} + \sigma_{1l}^2)$  ▷ Compute posterior second moments.
9:      $\bar{\boldsymbol{\rho}} \leftarrow \bar{\boldsymbol{\rho}}_l - \mathbf{X}^\top \mathbf{X} \bar{\mathbf{b}}_l$  ▷ Update expected residuals.
10:  end for
11:   $\sigma^2 \leftarrow \frac{1}{N} \text{ERSS-ss}(\mathbf{X}^\top \mathbf{X}, \mathbf{X}^\top \mathbf{y}, \mathbf{y}^\top \mathbf{y}, \bar{\mathbf{b}}_1, \dots, \bar{\mathbf{b}}_l, \bar{\mathbf{b}}_1^2, \dots, \bar{\mathbf{b}}_l^2)$  ▷ Optional; see 41.
12: until convergence criterion is satisfied
return  $\boldsymbol{\alpha}_1, \boldsymbol{\mu}_{11}, \sigma_{11}^2, \dots, \boldsymbol{\alpha}_L, \boldsymbol{\mu}_{1L}, \sigma_{1L}^2$ .
```

Likewise, the likelihood for σ_0^2, σ^2 under the SER model can be computed using only the sufficient statistics since it can be expressed as a weighted sum of the BF_s:

$$\begin{aligned}
\ell_{\text{SER}}(\sigma^2, \sigma_0^2; \mathbf{X}, \mathbf{y}) &:= p(\mathbf{y} \mid \mathbf{X}, \sigma^2, \sigma_0^2) \\
&= p(\mathbf{y} \mid \mathbf{X}, \sigma^2, b = 0) \sum_{j=1}^J \pi_j \text{BF}_j \\
&= \mathcal{N}_N(\mathbf{y}; \mathbf{0}, \sigma^2 \mathbf{I}_N) \sum_{j=1}^J \pi_j \text{BF}_j \\
&:= \ell_{\text{SER-ss}}(\sigma^2, \sigma_0^2; \mathbf{X}^\top \mathbf{X}, \mathbf{X}^\top \mathbf{y}). \tag{40}
\end{aligned}$$

Following [1], we compute the maximum-likelihood estimate of σ_0^2 by maximizing this likelihood via numerical optimization.

The IBSS-ss algorithm

The IBSS-ss algorithm is given in Algorithm 1. Additional notation used in Algorithm 1 includes: $\bar{\mathbf{b}}_l$, the expected value of \mathbf{b}_l with respect to the approximate posterior distribution, $q(\mathbf{b})$; and $\bar{\mathbf{b}}_l^2 = (b_{l1}^2, \dots, b_{lJ}^2)^\top$, the vector of posterior second moments $b_{lj}^2 := \mathbb{E}_q[b_{lj}^2]$. A key change in implementation is that the original IBSS algorithm keeps track of the posterior mean residuals $\bar{\mathbf{r}} := \mathbb{E}_q[\mathbf{X}^\top \mathbf{y} - \mathbf{b}] = \mathbf{X}^\top \mathbf{y} - \sum_{l=1}^L \bar{\mathbf{b}}_l$, whereas IBSS-ss updates $\bar{\boldsymbol{\rho}} := \mathbf{X}^\top \bar{\mathbf{r}}$. See [1] for development and justification for the IBSS algorithm, and details of implementation, including preprocessing steps, and calculation of the credible sets.

The only missing piece to the IBSS-ss algorithm is the expression for the expected residual sum of squares (ERSS) under the variational approximation to the posterior, $q(\mathbf{b})$, which is needed to estimate σ^2 . Again, the expression can be written in terms of

the sufficient statistics:

$$\begin{aligned}
& \text{ERSS}(\mathbf{X}, \mathbf{y}, \bar{\mathbf{b}}_1, \dots, \bar{\mathbf{b}}_L, \bar{\mathbf{b}}_1^2, \dots, \bar{\mathbf{b}}_L^2) \\
& := \mathbb{E}_q[\|\mathbf{y} - \mathbf{X}\mathbf{b}\|^2] \\
& = \|\mathbf{y} - \mathbf{X}\bar{\mathbf{b}}\|^2 - \sum_{l=1}^L \bar{\mathbf{b}}_l \mathbf{X}^\top \mathbf{X} \bar{\mathbf{b}}_l + \sum_{l=1}^L \sum_{j=1}^J (\mathbf{x}_j^\top \mathbf{x}_j) \bar{b}_{lj}^2 \\
& = \mathbf{y}^\top \mathbf{y} - 2\bar{\mathbf{b}}^\top \mathbf{X}^\top \mathbf{y} + \bar{\mathbf{b}}^\top \mathbf{X}^\top \mathbf{X} \bar{\mathbf{b}} - \sum_{l=1}^L \bar{\mathbf{b}}_l^\top \mathbf{X}^\top \mathbf{X} \bar{\mathbf{b}}_l + \sum_{l=1}^L \sum_{j=1}^J (\mathbf{x}_j^\top \mathbf{x}_j) \bar{b}_{lj}^2 \\
& := \text{ERSS-ss}(\mathbf{X}^\top \mathbf{X}, \mathbf{X}^\top \mathbf{y}, \mathbf{y}^\top \mathbf{y}, \bar{\mathbf{b}}_1, \dots, \bar{\mathbf{b}}_L, \bar{\mathbf{b}}_1^2, \dots, \bar{\mathbf{b}}_L^2). \tag{41}
\end{aligned}$$

The second ERSS definition makes explicit that it is a function of the sufficient statistics.

Computing sufficient statistics, and approximations, from summary data

Lemma 1 (Computing sufficient statistics). The statistics $\mathbf{X}^\top \mathbf{X}$, $\mathbf{X}^\top \mathbf{y}$ can be computed from the the summary data $\hat{\mathbf{b}}, \hat{\mathbf{s}}, \mathbf{R}, \mathbf{y}^\top \mathbf{y}, N$ from the following formulae:

$$\hat{z}_j = \hat{b}_j / \hat{s}_j \tag{42}$$

$$\hat{\sigma}_j^2 := \|\mathbf{y} - \hat{b}_j \mathbf{x}_j\|^2 / N = \frac{\mathbf{y}^\top \mathbf{y}}{\hat{z}_j^2 + N} \tag{43}$$

$$\mathbf{x}_j^\top \mathbf{y} = \hat{\sigma}_j^2 \hat{b}_j / \hat{s}_j^2 \tag{44}$$

$$d_j := \mathbf{x}_j^\top \mathbf{x}_j = \hat{\sigma}_j^2 / \hat{s}_j^2 \tag{45}$$

$$\mathbf{D}_{xx} = \text{diag}(d_1, \dots, d_J) \tag{46}$$

$$\mathbf{X}^\top \mathbf{X} = \mathbf{D}_{xx}^{1/2} \mathbf{R} \mathbf{D}_{xx}^{1/2}. \tag{47}$$

Proof. Equation 42 is the definition of the z -score. From the definitions of \hat{b}_j and \hat{s}_j (see Equations 4 and 5 in the main text), we have

$$\hat{z}_j^2 = N \times \frac{a_j}{1 - a_j},$$

in which

$$a_j = \hat{b}_j^2 \times \frac{\mathbf{x}_j^\top \mathbf{x}_j}{\mathbf{y}_j^\top \mathbf{y}_j}.$$

From this, the following identity holds:

$$\hat{z}_j^2 + N = \frac{N}{1 - a_j}.$$

Therefore, we have

$$\begin{aligned}
\hat{\sigma}_j^2 & := \|\mathbf{y} - \hat{b}_j \mathbf{x}_j\|^2 / N \\
& = \mathbf{y}^\top \mathbf{y} \times \frac{1 - a_j}{N} \\
& = \frac{\mathbf{y}^\top \mathbf{y}}{\hat{z}_j^2 + N},
\end{aligned}$$

which is (43).

Now, from the definition of \hat{s}_j (equation 5 in the main text), we have $\hat{s}_j^2 = \hat{\sigma}_j^2 / (\mathbf{x}_j^\top \mathbf{x}_j)$. This yields (44) because $\mathbf{x}_j^\top \mathbf{y} = \mathbf{x}_j^\top \mathbf{x}_j \hat{b}_j$, and (45). The remaining expressions follow from the definitions of \mathbf{D}_{xx} and $\hat{\mathbf{R}}$ in the main text (equation 7). \square

Remark 2. The formulae (43–47) are ordered in such a way that to provide a step-by-step procedure for reconstructing $\mathbf{X}^\top \mathbf{X}$, $\mathbf{X}^\top \mathbf{y}$ from the summary data $\hat{\mathbf{b}}, \hat{\mathbf{s}}, \mathbf{R}, \mathbf{y}^\top \mathbf{y}, N$. Thus, given $\hat{\mathbf{b}}, \hat{\mathbf{s}}, \mathbf{R}, \mathbf{y}^\top \mathbf{y}, N$, *SuSiE-RSS* involves first computing $\mathbf{X}^\top \mathbf{X}$, $\mathbf{X}^\top \mathbf{y}$ using (43–47), then applying the IBSS-ss algorithm to $\mathbf{X}^\top \mathbf{X}$, $\mathbf{X}^\top \mathbf{y}$, $\mathbf{y}^\top \mathbf{y}, N$.

Remark 3. Often, the in-sample LD matrix \mathbf{R} is not available, and is replaced with an estimate $\hat{\mathbf{R}}$. Given $\hat{\mathbf{b}}, \hat{\mathbf{s}}, \hat{\mathbf{R}}, \mathbf{y}^\top \mathbf{y}, N$, *SuSiE-RSS* involves first computing $\mathbf{V}_{xx} = \mathbf{X}^\top \mathbf{X} / N$, $\mathbf{X}^\top \mathbf{y}$ from (43–47), with $\hat{\mathbf{R}}$ replacing \mathbf{R} in the formula for $\mathbf{X}^\top \mathbf{X}$ to yield $N\mathbf{V}_{xx}$, then applying the IBSS-ss algorithm to $N\mathbf{V}_{xx}$, $\mathbf{X}^\top \mathbf{y}$, with $\sigma^2 = \mathbf{y}^\top \mathbf{y} / N$. Note that when σ^2 is fixed, the IBSS-ss algorithm does not need $\mathbf{y}^\top \mathbf{y}$ or N .

Remark 4. Simpler formulae for recovering summary statistics $\mathbf{X}^\top \mathbf{X}$, $\mathbf{X}^\top \mathbf{y}$ when \mathbf{X}, \mathbf{y} are standardized are given in the main text (“Special case when \mathbf{X}, \mathbf{y} are standardized”).

Dealing with uncentered data

As stated above, \mathbf{y} and the columns of \mathbf{X} are assumed to be centered, and accordingly the sufficient statistics should be computed using a centered \mathbf{X}, \mathbf{y} . If the sufficient statistics have been computed from an \mathbf{X}, \mathbf{y} that have not been centered, the sufficient statistics can be modified after the fact so that they correspond to a centered \mathbf{X} and \mathbf{y} . Denoting the unmodified data as \mathbf{X}, \mathbf{y} and the centered data as $\tilde{\mathbf{X}} := \mathbf{X} - \mathbf{1}_N \bar{\mathbf{x}}^\top$, $\tilde{\mathbf{y}} := \mathbf{y} - \bar{y} \mathbf{1}_N$, where $\bar{\mathbf{x}} = \mathbf{X}^\top \mathbf{1}_N / N$ is the vector of column means, $\bar{y} := \sum_{i=1}^N y_i / N$, and $\mathbf{1}_N$ is a column vector of ones of length N , the centering calculations for the sufficient statistics are

$$\begin{aligned}\tilde{\mathbf{X}}^\top \tilde{\mathbf{X}} &= \mathbf{X}^\top \mathbf{X} - N \bar{\mathbf{x}} \bar{\mathbf{x}}^\top \\ \tilde{\mathbf{X}}^\top \tilde{\mathbf{y}} &= \mathbf{X}^\top \mathbf{y} - N \bar{y} \bar{\mathbf{x}} \\ \tilde{\mathbf{y}}^\top \tilde{\mathbf{y}} &= \mathbf{y}^\top \mathbf{y} - N \bar{y}^2.\end{aligned}$$

Note if \mathbf{X}, \mathbf{y} are already centered, these operations will leave the summary statistics unchanged.

Similarly, the sufficient statistics may also be modified *post hoc* to be as if they were computed using a column-standardized \mathbf{X} ; that is, an \mathbf{X} in which each column has unit variance. Denoting the standardized (and centered) matrix as $\hat{\mathbf{X}}$, and assuming \mathbf{X} is centered, the calculations are

$$\begin{aligned}\hat{\mathbf{X}}^\top \hat{\mathbf{X}} &= N \times \mathbf{D}_{xx}^{-1/2} \mathbf{X}^\top \mathbf{X} \mathbf{D}_{xx}^{-1/2} \\ \hat{\mathbf{X}}^\top \mathbf{y} &= \sqrt{N} \times \mathbf{D}_{xx}^{-1/2} \mathbf{X}^\top \mathbf{y} \\ \mathbf{D}_{xx} &= \text{diag}((\mathbf{X}^\top \mathbf{X})_{11}, \dots, (\mathbf{X}^\top \mathbf{X})_{JJ}).\end{aligned}$$

Note that if \mathbf{X}, \mathbf{y} are already standardized, these operations will leave the summary statistics unchanged.

Alternative model-based derivation

We derived *SuSiE-RSS* by writing down the full-data likelihood in terms of sufficient statistics, approximating the sufficient statistics, then substituting the approximations to obtain an approximate likelihood (12). In the main text, we mentioned that, in the

standardized case, the resulting approximate likelihood can also be obtained from (22). Here we derive the more general result when \mathbf{X}, \mathbf{y} are not necessarily standardized.

Recall, the model for individual-level data is

$$\mathbf{y} \sim \mathcal{N}_J(\mathbf{X}\mathbf{b}, \sigma^2 \mathbf{I}_N). \quad (48)$$

From this, we have

$$\mathbf{X}^\top \mathbf{y} / \sqrt{N} \sim \mathcal{N}_J(\mathbf{X}^\top \mathbf{X} / N \times \sqrt{N} \mathbf{b}, \sigma^2 \mathbf{X}^\top \mathbf{X} / N). \quad (49)$$

If $\mathbf{X}^\top \mathbf{X}$ is invertible, the density obtained from (49) yields the same likelihood (up to a constant of proportionality) as the likelihood from the individual-level data (8). Substituting $\mathbf{X}^\top \mathbf{X}$ with $N\mathbf{V}_{xx}$ yields

$$\mathbf{X}^\top \mathbf{y} / \sqrt{N} \sim \mathcal{N}_J(\mathbf{V}_{xx} \times \sqrt{N} \mathbf{b}, \sigma^2 \mathbf{V}_{xx}). \quad (50)$$

If \mathbf{V}_{xx} is invertible, the density obtained from (50) yields the same likelihood (up to a constant of proportionality) as (12).

In the special case where \mathbf{X} and \mathbf{y} are standardized, these expressions simplify. Making substitutions $\mathbf{X}^\top \mathbf{y} / \sqrt{N} = \tilde{\mathbf{z}}$ and $\mathbf{X}^\top \mathbf{X} / N = \mathbf{R}$ in (49) results in the simplified expression

$$\tilde{\mathbf{z}} \sim \mathcal{N}_J(\sqrt{N} \mathbf{R} \mathbf{b}, \sigma^2 \mathbf{R}). \quad (51)$$

Finally, substituting $\hat{\mathbf{R}}$ for \mathbf{R} and fixing $\sigma^2 = 1$ gives (22).

Dealing with non-invertible LD matrix

If $\hat{\mathbf{R}}$ is not invertible, the models (20) and (22) do not have a density (with respect to Lebesgue measure). This complicates defining a likelihood from these models. Various approaches have been proposed for dealing with this issue. Some of these approaches are equivalent to using the likelihood (19) that we use here, which exists whether or not $\hat{\mathbf{R}}$ is invertible, while other approaches are not equivalent. Here we argue that using the likelihood (19) has the advantage of being simple, and satisfies the desirable property that inference under the SER model is independent of LD. We call this property ‘‘Irrelevance of null SNPs.’’

Note that the issues discussed here remain the same whether one uses models and likelihoods for the observed z -scores, $\hat{\mathbf{z}}$, or the PVE-adjusted z -scores, $\tilde{\mathbf{z}}$. Therefore, to simplify presentation we describe the results for the z -scores, noting that one can substitute for the PVE-adjusted z -scores wherever $\hat{\mathbf{z}}$ appears in the mathematical expressions below. To further simplify, we use $\mathbf{z} := \sqrt{N} \mathbf{b}$ to denote the scaled parameters, and compare the model

$$\hat{\mathbf{z}} \sim \mathcal{N}_J(\hat{\mathbf{R}} \mathbf{z}, \hat{\mathbf{R}}) \quad (52)$$

and the likelihood

$$\ell_{RSS-Z}(\mathbf{z}) := \exp(-\frac{1}{2} \mathbf{z}^\top \hat{\mathbf{R}} \mathbf{z} + \mathbf{z}^\top \hat{\mathbf{z}}), \quad (53)$$

which is the same (up to a constant of proportionality) as (19), but with $\hat{\mathbf{z}}$ replacing $\tilde{\mathbf{z}}$ and \mathbf{z} replacing $\sqrt{N} \mathbf{b}$. If $\hat{\mathbf{R}}$ is invertible, the density of the model (52) yields likelihood (53). We call (52) the ‘‘RSS-Z model’’ and (53) the ‘‘RSS-Z likelihood’’. The key point of this section is to argue that under model (52) the likelihood (53) is the ‘‘correct’’ likelihood even if $\hat{\mathbf{R}}$ is not invertible.

We assume that although $\hat{\mathbf{R}}$ may not be invertible, it is nonetheless a valid covariance matrix. That is, it is symmetric and positive semidefinite, which implies that all its eigenvalues are non-negative. This is guaranteed so long as $\hat{\mathbf{R}}$ is a sample

correlation matrix. However, it may be violated if $\hat{\mathbf{R}}$ is obtained by modifying a sample correlation matrix, for example by setting small correlations to zero. Any $J \times J$ symmetric positive semidefinite matrix $\hat{\mathbf{R}}$ with rank $r \leq J$ has an eigenvalue decomposition of the form

$$\hat{\mathbf{R}} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^\top, \quad (54)$$

where $\mathbf{\Lambda}$ is an $r \times r$ diagonal matrix with the r positive eigenvalues $\lambda_1 \geq \lambda_2 \cdots \geq \lambda_r > 0$ along its diagonal, and \mathbf{Q} is a $J \times r$ matrix whose columns are the r eigenvectors corresponding to the r non-zero eigenvalues, and $\mathbf{Q}^\top\mathbf{Q} = \mathbf{I}_r$ [4].

One can modify $\hat{\mathbf{R}}$ to make it invertible by simply adding a small diagonal element. Indeed, for any $\lambda \in (0, 1)$, the matrix

$$\hat{\mathbf{R}}_\lambda := (1 - \lambda)\hat{\mathbf{R}} + \lambda\mathbf{I} \quad (55)$$

will be invertible.

When $\hat{\mathbf{R}}$ is not invertible, the *RSS-Z* model (52) becomes degenerate, which means that some values of $\hat{\mathbf{z}}$ become impossible. In particular, with probability one, $\hat{\mathbf{z}} \in \text{range}(\mathbf{Q})$; that is, $\hat{\mathbf{z}} = \mathbf{Q}\boldsymbol{\alpha}$ for some $\boldsymbol{\alpha}$.

Definition 1 (Consistency of $\hat{\mathbf{z}}$ with $\hat{\mathbf{R}}$). We say that $\hat{\mathbf{z}}$ is *consistent* with $\hat{\mathbf{R}}$ if $\hat{\mathbf{z}} \in \text{range}(\mathbf{Q})$. Otherwise, if $\hat{\mathbf{z}} \notin \text{range}(\mathbf{Q})$ we say $\hat{\mathbf{z}}$ is *inconsistent* with $\hat{\mathbf{R}}$.

Note that, if $\hat{\mathbf{R}}$ is invertible, then $\text{range}(\mathbf{Q}) = \mathbb{R}^J$, and so $\hat{\mathbf{z}}$ will always be consistent with $\hat{\mathbf{R}}$. Further, if $\hat{\mathbf{z}}$ was generated from the model (52), then it will be consistent with $\hat{\mathbf{R}}$ (with probability 1). However, in practice the model (52) is only an approximation, and so in practice $\hat{\mathbf{z}}$ may be inconsistent with $\hat{\mathbf{R}}$.

Now we consider four approaches to dealing with a non-invertible $\hat{\mathbf{R}}$.

- 1a. Substitute the non-invertible $\hat{\mathbf{R}}$ with the invertible matrix $\hat{\mathbf{R}}_\lambda$, for some small λ , in (52). This is the approach used in [5, 6]. The model becomes $\hat{\mathbf{z}} \sim \mathcal{N}_J(\hat{\mathbf{R}}_\lambda \mathbf{z}, \hat{\mathbf{R}}_\lambda)$, which has a density because $\hat{\mathbf{R}}_\lambda$ is invertible, yielding likelihood

$$\ell_{1a}(\mathbf{z}; \hat{\mathbf{z}}, \hat{\mathbf{R}}, \lambda) := \exp(-\frac{1}{2}\mathbf{z}^\top \hat{\mathbf{R}}_\lambda \mathbf{z} + \mathbf{z}^\top \hat{\mathbf{z}}). \quad (56)$$

- 1b. Use the *RSS-Z* likelihood (53) even though $\hat{\mathbf{R}}$ is non-invertible, so the likelihood is

$$\ell_{1b}(\mathbf{z}; \hat{\mathbf{z}}, \hat{\mathbf{R}}) := \exp(-\frac{1}{2}\mathbf{z}^\top \hat{\mathbf{R}} \mathbf{z} + \mathbf{z}^\top \hat{\mathbf{z}}). \quad (57)$$

Note that the likelihood exists, and is easily computed, even when $\hat{\mathbf{R}}$ is not invertible. Earlier versions of FINEMAP [7] effectively used this approach (instead disallowing configurations $\gamma \subseteq \{1, \dots, J\}$ where $\hat{\mathbf{R}}_\gamma$ is not invertible).

- 2a. Set the covariance in the *RSS-Z* model (52) to $\hat{\mathbf{R}}_\lambda$ for some small λ , so that $\hat{\mathbf{z}} \sim \mathcal{N}_J(\hat{\mathbf{R}} \mathbf{z}, \hat{\mathbf{R}}_\lambda)$. Note that this approach differs from 1a because it uses $\hat{\mathbf{R}}$ instead of $\hat{\mathbf{R}}_\lambda$ for the mean. This yields the following likelihood:

$$\ell_{2a}(\mathbf{z}; \hat{\mathbf{z}}, \hat{\mathbf{R}}, \lambda) := \exp(-\frac{1}{2}\mathbf{z}^\top \hat{\mathbf{R}} \hat{\mathbf{R}}_\lambda^{-1} \hat{\mathbf{R}} \mathbf{z} + \mathbf{z}^\top \hat{\mathbf{R}} \hat{\mathbf{R}}_\lambda^{-1} \hat{\mathbf{z}}). \quad (58)$$

- 2b. Project $\hat{\mathbf{z}}$ into a lower-dimensional subspace, $\hat{\mathbf{z}}' := \mathbf{\Lambda}^{-1/2}\mathbf{Q}^\top \hat{\mathbf{z}}$, which ensures that $\hat{\mathbf{z}}' \in \mathbb{R}^r$ has a probability density, $\hat{\mathbf{z}}' \sim \mathcal{N}_r(\mathbf{\Lambda}^{1/2}\mathbf{Q}^\top \mathbf{z}, \mathbf{I}_r)$. Thus we have

$$\begin{aligned} p(\hat{\mathbf{z}}' | \mathbf{z}, \hat{\mathbf{R}}) &\propto \exp\{-\frac{1}{2}(\hat{\mathbf{z}}' - \mathbf{\Lambda}^{1/2}\mathbf{Q}^\top \mathbf{z})^\top (\hat{\mathbf{z}}' - \mathbf{\Lambda}^{1/2}\mathbf{Q}^\top \mathbf{z})\} \\ &\propto \exp(-\frac{1}{2}\mathbf{z}^\top \hat{\mathbf{R}} \mathbf{z} + \mathbf{z}^\top \mathbf{Q} \mathbf{\Lambda}^{1/2} \hat{\mathbf{z}}') \\ &= \exp(-\frac{1}{2}\mathbf{z}^\top \hat{\mathbf{R}} \mathbf{z} + \mathbf{z}^\top \mathbf{Q} \mathbf{Q}^\top \hat{\mathbf{z}}), \end{aligned} \quad (59)$$

and therefore the likelihood is

$$\ell_{2b}(\mathbf{z}; \hat{\mathbf{z}}, \hat{\mathbf{R}}) = \exp(-\frac{1}{2}\mathbf{z}^\top \hat{\mathbf{R}}\mathbf{z} + \mathbf{z}^\top \mathbf{Q}\mathbf{Q}^\top \hat{\mathbf{z}}). \quad (60)$$

The same likelihood is obtained by replacing $\hat{\mathbf{R}}^{-1}$ in the density for (52) with the pseudoinverse of $\hat{\mathbf{R}}$, which is $\mathbf{Q}\mathbf{\Lambda}^{-1}\mathbf{Q}^\top$. This is the approach used in msCAVIAR [8].

We summarize the connections between these four approaches in the following proposition.

Proposition 2. (a) As $\lambda \rightarrow 0$, Approach 1a becomes equivalent to Approach 1b, and Approach 2a becomes equivalent to Approach 2b; that is,

$$\lim_{\lambda \rightarrow 0} \ell_{1a}(\mathbf{z}; \hat{\mathbf{z}}, \hat{\mathbf{R}}, \lambda) = \ell_{1b}(\mathbf{z}; \hat{\mathbf{z}}, \hat{\mathbf{R}}); \quad (61)$$

$$\lim_{\lambda \rightarrow 0} \ell_{2a}(\mathbf{z}; \hat{\mathbf{z}}, \hat{\mathbf{R}}, \lambda) = \ell_{2b}(\mathbf{z}; \hat{\mathbf{z}}, \hat{\mathbf{R}}). \quad (62)$$

(b) Approaches 1b and 2b are equivalent—*i.e.*, $\ell_{1b}(\mathbf{z}; \hat{\mathbf{z}}, \hat{\mathbf{R}}) = \ell_{2b}(\mathbf{z}; \hat{\mathbf{z}}, \hat{\mathbf{R}})$ —if and only if $\hat{\mathbf{z}}$ is consistent with $\hat{\mathbf{R}}$.

(c) If $\hat{\mathbf{z}}$ is inconsistent with $\hat{\mathbf{R}}$, Approach 2b behaves discontinuously,

$$\lim_{\lambda \rightarrow 0} \ell_{2b}(\mathbf{z}; \hat{\mathbf{z}}, \hat{\mathbf{R}}_\lambda) \neq \ell_{2b}(\mathbf{z}; \hat{\mathbf{z}}, \hat{\mathbf{R}}), \quad (63)$$

but in the limit it is equivalent to Approach 1b,

$$\lim_{\lambda \rightarrow 0} \ell_{2b}(\mathbf{z}; \hat{\mathbf{z}}, \hat{\mathbf{R}}_\lambda) = \ell_{1b}(\mathbf{z}; \hat{\mathbf{z}}, \hat{\mathbf{R}}). \quad (64)$$

Proof. (a) As $\lambda \rightarrow 0$, $\hat{\mathbf{R}}_\lambda \rightarrow \hat{\mathbf{R}}$, so (61) is trivially satisfied. To prove (62), we define $\mathbf{B} := \mathbf{Q}\mathbf{\Lambda}^{1/2}$ with pseudoinverse $\mathbf{B}^\dagger = \mathbf{\Lambda}^{-1/2}\mathbf{Q}^\top$. With these definitions, we can write $\ell_{2a}(\mathbf{z}; \hat{\mathbf{z}}, \hat{\mathbf{R}}, \lambda)$ as

$$\begin{aligned} \ell_{2a}(\mathbf{z}; \hat{\mathbf{z}}, \hat{\mathbf{R}}, \lambda) &= \exp(-\frac{1}{2}\mathbf{z}^\top \mathbf{B}\mathbf{B}^\top((1-\lambda)\mathbf{B}\mathbf{B}^\top + \lambda\mathbf{I})^{-1}\mathbf{B}\mathbf{B}^\top\mathbf{z} \\ &\quad + \mathbf{z}^\top \mathbf{B}\mathbf{B}^\top((1-\lambda)\mathbf{B}\mathbf{B}^\top + \lambda\mathbf{I})^{-1}\hat{\mathbf{z}}). \end{aligned} \quad (65)$$

In the limit as $\lambda \rightarrow 0$, $\mathbf{B}^\top((1-\lambda)\mathbf{B}\mathbf{B}^\top + \lambda\mathbf{I})^{-1} \rightarrow \mathbf{B}^\dagger$ (Theorem 3.4 in [9]). Therefore,

$$\begin{aligned} \lim_{\lambda \rightarrow 0} \ell_{2a}(\mathbf{z}; \hat{\mathbf{z}}, \hat{\mathbf{R}}, \lambda) &= \exp(-\frac{1}{2}\mathbf{z}^\top \mathbf{B}\mathbf{B}^\dagger\mathbf{B}\mathbf{B}^\top\mathbf{z} + \mathbf{z}^\top \mathbf{B}\mathbf{B}^\dagger\hat{\mathbf{z}}) \\ &= \exp(-\frac{1}{2}\mathbf{z}^\top \hat{\mathbf{R}}\mathbf{z} + \mathbf{z}^\top \mathbf{Q}\mathbf{Q}^\top \hat{\mathbf{z}}) \\ &= \ell_{2b}(\mathbf{z}; \hat{\mathbf{z}}, \hat{\mathbf{R}}). \end{aligned} \quad (66)$$

(b) $\ell_{1b}(\mathbf{z}; \hat{\mathbf{z}}, \hat{\mathbf{R}}) = \ell_{2b}(\mathbf{z}; \hat{\mathbf{z}}, \hat{\mathbf{R}})$ for all \mathbf{z} if and only if $\mathbf{Q}\mathbf{Q}^\top \hat{\mathbf{z}} = \hat{\mathbf{z}}$. Since $\mathbf{P} = \mathbf{Q}\mathbf{Q}^\top$ is an orthogonal projector onto $\text{range}(\mathbf{Q})$, $\hat{\mathbf{z}} \in \text{range}(\mathbf{Q}) \Leftrightarrow \mathbf{Q}\mathbf{Q}^\top \hat{\mathbf{z}} = \hat{\mathbf{z}}$ (see [4], Ch. 6).

(c) First we prove (64). Since $\hat{\mathbf{R}}_\lambda$ is full rank, the $J \times J$ matrix of eigenvectors \mathbf{Q}_λ satisfies $\mathbf{Q}_\lambda \mathbf{Q}_\lambda^\top = \mathbf{I}_J$. Therefore,

$$\ell_{2b}(\mathbf{z}; \hat{\mathbf{z}}, \hat{\mathbf{R}}_\lambda) = \exp(-\frac{1}{2}\mathbf{z}^\top \hat{\mathbf{R}}_\lambda \mathbf{z} + \mathbf{z}^\top \hat{\mathbf{z}}) = \ell_{1a}(\mathbf{z}; \hat{\mathbf{z}}, \hat{\mathbf{R}}_\lambda), \quad (67)$$

so the result follows from (61). The result (63) then follows from part (b). \square

Remark 5. Proposition 2 (a) and (b) together imply that if $\hat{\mathbf{z}}$ is consistent with $\hat{\mathbf{R}}$ then, for sufficiently small λ , all four approaches should give the same results (ignoring numerical errors that occur in floating-point computations). However, because the RSS-Z model is only an approximation, $\hat{\mathbf{z}}$ will often be inconsistent with $\hat{\mathbf{R}}$. It is this fact that causes the methods to give different results.

Irrelevance of null SNPs

Given that the different approaches to dealing with non-invertible $\hat{\mathbf{R}}$ may give different results, it is natural to ask which approach is preferable. Here we argue that Approaches 1a and 1b are preferable to Approaches 2a and 2b because Approaches 1a and 1b always satisfy a simple property that we call “irrelevance of null SNPs.” (Approaches 2a and 2b may sometimes satisfy this property, but they are not guaranteed to do so.) This property is also satisfied by the full data likelihood and implies, among other things, that inference under the SER model is independent of the LD matrix.

We motivate this property by observing that the individual-data multiple regression likelihood (1) has the following simple property: if a SNP j has no effect (a “null SNP”), then its genotypes \mathbf{x}_j do not appear in the likelihood. As a result, genotypes at null SNPs do not impact inference for other SNPs. We call this property “irrelevance of null SNPs.”

In the summary-data setting, we do not directly observe the genotypes, so we need to formulate an analogous property. Therefore, to translate these ideas to the summary-data likelihoods with $\mathbf{z}, \hat{\mathbf{R}}$, we ask instead whether the likelihood has the following property: if $z_j = 0$, then $\hat{R}_{j1}, \dots, \hat{R}_{jJ}$ do not appear in the likelihood. If, for example, $\hat{R}_{jj'}$ is the correlation between SNPs j and j' obtained from a suitable reference panel, then this property implies that the genotypes for SNP j have no impact on the summary-data likelihood when $z_j = 0$. (The same can be said for regularized LD matrices of the form (23).)

To formalize these ideas, we introduce some notation. We use γ to denote a subset of SNPs, $\gamma \subseteq \{1, \dots, J\}$, and we let \mathbf{z}_γ denote the elements of the vector \mathbf{z} corresponding to the SNPs in γ . The remaining elements are denoted by $\mathbf{z}_{-\gamma}$. Similarly, we use $\hat{\mathbf{R}}_\gamma$ to denote the $|\gamma| \times |\gamma|$ matrix formed by the rows and columns of $\hat{\mathbf{R}}$ corresponding to elements of γ , and $\hat{\mathbf{R}}_{-\gamma}$ denotes the matrix formed by all rows and columns of $\hat{\mathbf{R}}$ that do not correspond to elements of γ . We then define irrelevance of null SNPs as follows.

Definition 2 (Irrelevance of null SNPs). Let $\ell(\mathbf{z})$ be any likelihood for \mathbf{z} (which implicitly depends on $\hat{\mathbf{z}}, \hat{\mathbf{R}}$, and the model parameters). For any subset $\gamma \subseteq \{1, \dots, J\}$, let $\ell^\gamma(\mathbf{z}_\gamma)$ denote the likelihood for \mathbf{z}_γ when the remaining elements $\mathbf{z}_{-\gamma}$ are set to zero; that is,

$$\ell^\gamma(\mathbf{z}_\gamma) := \ell(\mathbf{z}_\gamma, \mathbf{z}_{-\gamma} = \mathbf{0}). \quad (68)$$

We say the likelihood $\ell(\mathbf{z})$ satisfies the irrelevance of null SNPs property if, for all γ , $\ell^\gamma(\mathbf{z}_\gamma)$ depends on $\hat{\mathbf{R}}$ only through $\hat{\mathbf{R}}_\gamma$.

Remark 6. We have framed the definition in terms of likelihoods for the scaled parameters \mathbf{z} , but a similar definition could be obtained for the effects \mathbf{b} by replacing \mathbf{z} with \mathbf{b} .

The multiple regression likelihood based on individual-level data (8) satisfies the irrelevance of null SNPs property. Likelihoods ℓ_{1a} (56) and ℓ_{1b} (57) also satisfy this property, as summarized by the following proposition.

Proposition 3. The ℓ_{RSS-Z} likelihood (53) satisfies irrelevance of null SNPs (Definition 2).

Proof. Setting $\mathbf{z}_{-\gamma} = \mathbf{0}$ in (53) yields

$$\ell_{RSS-Z}(\mathbf{z}_\gamma, \mathbf{z}_{-\gamma} = \mathbf{0}) = \exp\left(-\frac{1}{2}\mathbf{z}_\gamma^\top \hat{\mathbf{R}}_\gamma \mathbf{z}_\gamma + \mathbf{z}_\gamma^\top \hat{\mathbf{z}}_\gamma\right). \quad (69)$$

□

The irrelevance of null SNPs has the following simple implication: to assess support for the hypothesis $H_\gamma : \mathbf{z}_\gamma = \mathbf{0}$, one only needs the genotypes corresponding to the non-null SNPs. Indeed, if $p_\gamma(\mathbf{z}_\gamma)$ denotes any prior on \mathbf{z}_γ under H_γ then, in the absence of nuisance parameters, the Bayes Factor for H_γ vs. H_0 is

$$\text{BF}_\gamma := \frac{\int \ell^\gamma(\mathbf{z}_\gamma) p_\gamma(\mathbf{z}_\gamma) d\mathbf{z}_\gamma}{\ell^\gamma(\mathbf{z}_\gamma = \mathbf{0})}. \quad (70)$$

This result implies that BF_γ depends only on $\hat{\mathbf{z}}_\gamma, \hat{\mathbf{R}}_\gamma$ whatever priors are used for each γ (assuming that the prior p_γ does not depend on the null genotypes). This result is easily extended to integrate out additional nuisance parameters (e.g., σ^2 in the multiple regression model) in both the numerator and denominator.

A similar result is shown for specific priors in [7], and is exploited in FINEMAP. Our analysis here emphasizes that this is, fundamentally, due to properties of the likelihood, and is not confined to specific priors.

Remark 7. Applying this result to the special case that γ contains a single SNP j , i.e., $\gamma = \{j\}$, BF_γ depends on the genotypes only through SNP j ; in particular, BF_γ does not depend on the LD between SNPs. Thus, irrelevance of null SNPs implies that fitting a SER does not depend on LD. (A similar observation has been made for prospective models of case-control traits based on conditional-independence arguments [10]). Since the likelihood (13) satisfies irrelevance of null SNPs, inference under the SER model with this likelihood has the desirable property that it does not depend on LD.

Estimation of λ in regularized LD matrix

To solve (24), we used the Brent-Dekker algorithm [11], which is implemented in R by the `optimize` function. This algorithm performs a 1-d search over $\lambda \in [0, 1]$. The main computational expense of this optimization step is the eigenvalue decomposition of $\hat{\mathbf{R}}_0$. Computing the eigenvalue decomposition has computational complexity $O(J^3)$, and therefore can impose a substantial computational burden on the overall fine-mapping analysis when J is large. In practice, we found that the regularization typically provided only a small improvement to the *SuSiE* fine-mapping results, so in the software we set $\lambda = 0$ by default to avoid this potentially large computational expense.

Likelihood ratio for detecting allele flips

Based on the conditional distribution (26), we developed a likelihood ratio to detect allele flips. Consider that, when $\hat{\mathbf{R}} = \mathbf{R}$, the standardized differences (27) should be approximately standard normal; $t_j \sim \mathcal{N}(0, 1)$. However, when $\hat{\mathbf{R}}$ is estimated from a reference panel, even without errors such as allele flips, the empirical distribution of standardized differences will be longer tailed than the standard normal. This suggests that a more flexible distribution should be used to model the standardized differences. We used a mixture of normals to model the empirical conditional distribution,

$$\tilde{z}_j \mid \tilde{\mathbf{z}}_{-j}, \hat{\mathbf{R}} \sim \sum_{k=1}^K w_k \mathcal{N}(-\boldsymbol{\Omega}_{j,-j} \tilde{\mathbf{z}}_{-j} / \Omega_{jj}, \sigma_k^2 / \Omega_{jj}), \quad (71)$$

where $\sigma_1, \dots, \sigma_K$ are pre-specified standard deviations such that $\sigma_1 < \dots < \sigma_K$, and $\mathbf{w} = (w_1, \dots, w_K)$ are mixture proportions (non-negative and sum to 1). In our analyses, we chose the σ_k 's such that $\sigma_1 = 0.8$, $\sigma_K = 2 \times \max\{|t_1|, \dots, |t_J|\}$, and $\sigma_{k+1} = 1.05 \times \sigma_k$. We estimated \mathbf{w} by maximum likelihood, using summary data for all

SNPs. Computing the maximum-likelihood estimate of \mathbf{w} is a convex optimization problem and can be solved efficiently using `mixsqp` [12]. (When J is large, the effort involved in solving this convex optimization problem is negligible compared to inverting or factorizing $\hat{\mathbf{R}}$.) We then used the maximum-likelihood estimates of the mixture weights, $\hat{\mathbf{w}} = (\hat{w}_1, \dots, \hat{w}_K)$, to compute a likelihood ratio for each SNP j ,

$$\text{LR}_j := \frac{\sum_{k=1}^K \hat{w}_k \mathcal{N}(\tilde{z}_j; \mathbf{\Omega}_{j,-j} \tilde{\mathbf{z}}_{-j} / \Omega_{jj}, \sigma_k^2 / \Omega_{jj})}{\sum_{k=1}^K \hat{w}_k \mathcal{N}(\tilde{z}_j; -\mathbf{\Omega}_{j,-j} \tilde{\mathbf{z}}_{-j} / \Omega_{jj}, \sigma_k^2 / \Omega_{jj})}. \quad (72)$$

This likelihood ratio can only identify errors in SNPs j with z -scores that are large in magnitude. Therefore, after estimating the mixture weights, we focus on SNPs j with $|\tilde{z}_j| > 2$.

To verify the use of this likelihood ratio to identify allele flips, we simulated fine-mapping data sets in which exactly one SNP was a causal SNP, and exactly one of the SNPs had a flipped allele—that is, the allele encoding used to compute the z -scores was different from of the allele encoding used to compute the LD matrix. We considered two scenarios in these simulations: (1) the allele-flip SNP was also the causal SNP; (2) the allele-flip SNP was not the causal SNP. An allele flip in both of these scenarios can affect accuracy of the fine-mapping so it is important to identify and eliminate the allele-flip SNP in both cases.

We simulated fine-mapping data sets using the UK Biobank genotypes, as previously described in the Results (see also “Details of simulations” below), with the following changes: we simulated one causal SNP with an effect size chosen so that it explained 2% of variance in \mathbf{y} ; and we used 10,000 samples to compute the z -scores and LD matrix. The z -scores and LD matrix were computed using the same samples (“in-sample LD”), except that the LD matrix was computed only after modifying the genotypes of the allele-flip SNP. Once the summary data $\hat{\mathbf{z}}, \hat{\mathbf{R}}$ were computed from the individual-level data, we computed LR_j for each SNP j . (We computed the likelihood ratios using the unadjusted z -scores, \hat{z}_j , instead of the PVE-adjusted z -scores, \tilde{z}_j . The latter is recommended, but here this would have made little difference because the sample size was large and the SNP effects were small.) In total, we simulated 200 data sets for the first scenario and another 200 data sets for the second scenario.

The results of these simulations are summarized in [S1 Fig](#). The allele-flip SNPs almost always had a likelihood ratio greater than 1 (*i.e.*, $\log \text{LR}_j > 0$), and other SNPs usually had likelihood ratios less than 1 (note the logarithmic scale in some of the plots). A small proportion of SNPs without an allele flip had likelihood ratios greater than 1 (plots in middle row). However, most of these SNPs had z -scores close to zero, so restricting to SNPs with larger z -scores (plots in bottom row) eliminates most of these false positives. In summary, based on these simulations, the likelihood-ratio statistic (72) provides an accurate diagnostic for identification of allele flips so long as we restrict attention to SNPs with larger z -scores.

***SuSiE* refinement procedure**

The refinement procedure is outlined in Algorithm 2. This procedure will work with any individual-level data or summary data accepted by *SuSiE* or *SuSiE-RSS*. In the algorithm, when we say the *SuSiE* model fit s is initialized to fit t , specifically we mean that the posterior means $\bar{\mathbf{b}}_l$ for each single effect $l = 1, \dots, L$ are initialized from t . (Refer to Algorithm 1 in [1], and Algorithm 1 in this paper.)

Algorithm 2 *SuSiE* refinement procedure

Require: A *SuSiE* model fit, s , with $K \geq 1$ credible sets, CS_1, \dots, CS_K .

```
1:  $F \leftarrow \text{ELBO}(s)$ 
2: repeat
3:   for  $k = 1$  to  $K$  do
4:      $\tilde{\pi} \leftarrow \pi$ 
5:      $\tilde{\pi}_j \leftarrow 0$  for all  $j \in CS_k$ 
6:     Fit SuSiE model,  $t_k$ , using prior weights  $\tilde{\pi}$  and default initialization
7:     Fit SuSiE model,  $s_k$ , using prior weights  $\pi$ , initialized at  $t_k$ 
8:      $F_k \leftarrow \text{ELBO}(s_k)$ 
9:   end for
10:   $k^* \leftarrow \text{argmax}_k \text{ELBO}(s_k)$ 
11:  if  $F_{k^*} > F$  then
12:     $s \leftarrow s_{k^*}$ 
13:  end if
14: until  $F_{k^*} \leq F$ 
```

Details of calculations for toy example

In the toy example (see “Fine-mapping with inconsistent summary data and a non-invertible LD matrix: an illustration” in the Results), we assumed $\hat{\mathbf{R}}$ is the 2×2 rank-1 matrix of all ones. Then the eigenvalue decomposition of $\hat{\mathbf{R}}$ is $\hat{\mathbf{R}} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^\top$ with $\mathbf{\Lambda} = 1$, $\mathbf{Q} = (\sqrt{1/2}, \sqrt{1/2})^\top$, and $\mathbf{Q}\mathbf{Q}^\top$ is the 2×2 matrix with all entries set to $1/2$. In the likelihood $\ell_{2b}(\mathbf{z}; \hat{\mathbf{z}}, \hat{\mathbf{R}})$ (see eq. 60), this has the effect that $\mathbf{Q}\mathbf{Q}^\top \hat{\mathbf{z}}$ is the average of the observed z -scores; that is, $\mathbf{Q}\mathbf{Q}^\top \hat{\mathbf{z}} = ((\hat{z}_1 + \hat{z}_2)/2, (\hat{z}_1 + \hat{z}_2)/2)^\top$.

The *SuSiE-RSS* results for this toy example with $\hat{\mathbf{z}} = (6, 7)$ were generated by running `susie_rss` with the default settings (`susieR` version 0.12.06).

Details of simulations

We evaluated the fine-mapping methods on summary data sets generated using real genotypes and simulated phenotypes. For genotype data, we used version 3 of the imputed genotypes from the UK Biobank resource [13]. UK Biobank is a large-scale biomedical database and research resource containing genetic, lifestyle and health information from half a million UK participants. The UK Biobank genotype data are well suited for fine-mapping because of their large sample size (approximately 500,000 genotyped individuals) and the high density of available genetic variants after genotype imputation [14].

Following [15], we took steps to filter out genotype samples, resulting in a candidate set of 274,549 samples. In detail, to limit confounding due to population structure, we considered only genotype samples marked as “White British” (based on a principal components analysis of the genotypes [14]). We then removed any samples from the “White British” subset that met one or more of the following criteria: mismatch between self-reported and genetic sex; outlier based on heterozygosity and/or rate of missing genotypes; has at least one close relative in the same data set (based on the UK Biobank’s kinship and “relatedness” calculations); or does not have a measurement of standing height.

From this collection of 274,549 genotype samples, we chose uniformly at a random a subset of $N = 50,000$ samples, then we used this random subset of genotypes to generate a collection of genotypes for fine-mapping. We repeated this 200 times to generate 200 genotype data sets for fine-mapping. Each of these 200 data sets contained SNPs within selected regions on autosomal chromosomes. These regions contained

roughly 1,000 SNPs, and were chosen so that no two regions contained the same SNPs. A SNP was included in a data set only if it satisfied all of the following criteria: SNP with at most two alleles; minor allele frequency of 1% or greater; and information score, which quantifies imputation quality, of 0.9 or greater. On average, 998 SNPs were included in a region. The smallest region contained 998 SNPs, and the largest contained 1,001 SNPs. The average size of a region in base pairs was 390 kb.

For each of the 200 randomly chosen regions, we generated three data sets by following a procedure similar to [1], resulting in a total of $200 \times 3 = 600$ fine-mapping data sets. Our procedure is briefly described here. We simulated phenotypes \mathbf{y} under the multiple regression model (1) in which \mathbf{X} was the centered and standardized matrix of 50,000 genotypes. We simulated three sets of phenotype data from the same \mathbf{X} by setting the number of causal SNPs to be 1, 2 or 3. The causal SNPs were chosen uniformly at random among the available SNPs in the region. The causal (non-zero) SNP effects b_j were drawn randomly from the standard normal, then the residual variance σ^2 was adjusted so that the genotypes at all SNPs in the region explained 0.5% of the variance in \mathbf{y} . The outcomes \mathbf{y} were then simulated as $y_i = x_{i1}b_1 + \dots + x_{iJ}b_J + \varepsilon_i$, with $\varepsilon_i \sim \mathcal{N}_N(0, \sigma^2)$. We then calculated z -scores, \mathbf{z} , and the in-sample LD matrix, \mathbf{R} , from \mathbf{X} and the simulated \mathbf{y} .

To investigate the impact of misspecification of the LD matrix, we randomly sampled subsets of $n = 500$ and $n = 1,000$ individuals (not overlapping with the $N = 50,000$ samples used to compute the z -scores), and computed two “out-of-sample” LD matrices, denoted by $\hat{\mathbf{R}}_{500}$ and $\hat{\mathbf{R}}_{1000}$, respectively. Because the sample sizes were not large (at most 50,000), it was feasible to compute all in-sample and out-of-sample LD matrices using the function `cor` in R.

To assess performance in identifying causal SNPs with larger effects, we repeated the same simulations as described above, except that (a) we used a smaller number of samples, and (b) σ^2 was adjusted so that the genotypes at all SNPs explained a larger proportion variance in \mathbf{y} . Specifically, we repeated simulations at target PVE settings of 10% and 30%, and for these settings we used $N = 2,500$ samples and $N = 800$ samples, respectively, to achieve roughly the same power as the simulations with 0.5% and $N = 50,000$ samples. The the out-of-sample LD matrix was obtained in each simulation using a non-overlapping subset of $n = 1,000$ individuals.

Fine-mapping methods

In the simulations, all *SuSiE-RSS* variants were implemented by calling function `susie_rss` from the R package `susieR` (version 0.12.06). Unless otherwise stated, we set the maximum number of non-zero effects to 10 ($L = 10$), we fixed the residual variance σ^2 to 1 (`estimate_residual_variance = FALSE`, `residual_variance = 1`), we set the maximum number of IBSS-ss iterations to 1,000 (`max_iter = 1000`), and we used refinement (`refine = TRUE`). In all calls to `susie_rss`, the summary data provided were the z -scores, LD matrix, and sample size (passed via `susie_rss` arguments \mathbf{z} , \mathbf{R} and n). We considered the following variants of *SuSiE-RSS*: `susie_rss` with L , the maximum number of non-zero effects in the regression model, set to 10 or the true value (1, 2 or 3); with and without the refinement procedure (`refine = TRUE`, `refine = FALSE`); and estimating the residual variance σ^2 or fixing it to 1 (`estimate_residual_variance = TRUE` or `estimate_residual_variance = FALSE`). Note that *SuSiE-RSS* with sufficient data (\mathbf{R} , $\hat{\mathbf{z}}$, N) and `estimate_residual_variance = TRUE` (second row of Table 1) gives the same result as *SuSiE* with individual-level data.

We also assessed performance of FINEMAP [7] (version 1.4), CAVIAR [5] (version 2.2) and DAP-G [16, 17] (git commit id 875ba40). All these methods accept summary data \mathbf{z} , $\hat{\mathbf{R}}$ as input. FINEMAP additionally requires N , the number of samples used to

calculate the z -scores. (DAP-G accepts either sufficient statistics or summary data, but in our simulations we only evaluated DAP-G with summary data $\mathbf{z}, \hat{\mathbf{R}}$.) These methods are based on the same multiple linear regression model as *SuSiE-RSS*, differing in the choice of priors, the approach used to compute posterior probabilities, and definition of a credible set.

We ran the FINEMAP program with flags `--sss --n-causal-snps 5`; with these options, FINEMAP used shotgun stochastic search to explore causal configurations, restricting to configurations with at most 5 causal SNPs (this is also the default setting in FINEMAP 1.4). For the “FINEMAP, $L = \text{true}$ ” results, we instead called FINEMAP with `--sss --n-causal-snps L`, where L was the number of causal SNPs used in the simulation (1, 2 or 3). In FINEMAP version 1.4, a credible set was defined conditioned on the number of causal SNPs, k : “For a specific k , FINEMAP takes the k -SNP causal configuration with highest posterior probability and then asks, for the l th SNP in that set, which are the other candidates that could possibly replace that SNP in this causal configuration. The l th credible set shows the best candidate SNPs and their posterior probability of being in a k -SNP causal configuration that additionally contains $k - 1$ SNPs. Note that the $k - 1$ SNPs are chosen to have highest posterior probability in their credible set.” FINEMAP outputted a set of results for each $k = 1, \dots, L$. We kept the credible sets from the k with the highest posterior probability.

We ran DAP-G using the default settings. Note that the default maximum number of causal SNPs in DAP-G (the “maximum model size”) is J , the total number of SNPs. The DAP-G software outputs “signal clusters” [16], not credible sets. However, we were able to compute credible sets from the DAP-G output following this description from [16]: “For a signal whose local $\text{fdr} \leq t$, it is straightforward to construct a $(1 - t)\%$ Bayesian credible set by selecting a minimum subset of SNPs such that their cumulative SNP-level PIPs reach $1 - t$.” We implemented this calculation in R to generate the DAP-G credible sets.

We ran CAVIAR with flags `-g 0.001 -c L` so that all SNPs had a prior inclusion probability of $1/1000$, and the maximum number of causal SNPs was L , where L was the value used to simulate the phenotypes (1, 2 or 3). The remaining CAVIAR parameters were kept at their default settings.

To compare CSs generated by *SuSiE-RSS*, FINEMAP and DAP-G (Fig. 4), we first filtered out all CSs with purity less than 0.5 (this is also the default setting in *susier*). Following [1], “purity” was defined as the smallest absolute correlation among all pairs of SNPs within a CS.

Computing environment

All simulations were run on Linux machines (Scientific Linux 7.4) with Intel Xeon E5-2680v4 (“Broadwell”) processors. *SuSiE-RSS* was run in R 3.6.1 [18] linked to the OpenBLAS 0.2.19 optimized numerical libraries. At most 2 GB of memory was needed to run *SuSiE-RSS* on the simulated data sets, and at most 10 GB was needed to run DAP-G, FINEMAP and CAVIAR. All methods and other computations were run without multithreading (one CPU). Runtime statistics for running the methods on summary data with in-sample LD matrices are given in Table 2.

We used the Dynamic Statistical Comparisons system (<https://github.com/stephenslab/dsc>) to perform the simulations. All code implementing the simulations, including the compiled results generated from our simulations, are available at https://github.com/stephenslab/dsc_susierss, and were deposited on Zenodo [19].

References

1. Wang G, Sarkar A, Carbonetto P, Stephens M. A simple new approach to variable selection in regression, with application to genetic fine mapping. *Journal of the Royal Statistical Society, Series B.* 2020;82(5):1273–1300.
2. Servin B, Stephens M. Imputation-based analysis of association studies: candidate regions and quantitative traits. *PLoS Genetics.* 2007;3(7):e114.
3. Chipman H, George EI, McCulloch RE. The practical implementation of Bayesian model selection. In: *Model Selection.* vol. 38 of IMS Lecture Notes. Institute of Mathematical Statistics; 2001. p. 65–116.
4. Trefethen LN, Bau D. *Numerical linear algebra.* Philadelphia, PA: SIAM; 1997.
5. Hormozdiari F, Kostem E, Kang EY, Pasaniuc B, Eskin E. Identifying causal variants at loci with multiple signals of association. *Genetics.* 2014;198(2):497–508.
6. Kichaev G, Yang WY, Lindstrom S, Hormozdiari F, Eskin E, Price AL, et al. Integrating functional data to prioritize causal variants in statistical fine-mapping studies. *PLoS Genetics.* 2014;10(10):e1004722.
7. Benner C, Spencer CCA, Havulinna AS, Salomaa V, Ripatti S, Pirinen M. FINEMAP: efficient variable selection using summary data from genome-wide association studies. *Bioinformatics.* 2016;32(10):1493–1501.
8. LaPierre N, Taraszka K, Huang H, He R, Hormozdiari F, Eskin E. Identifying causal variants by fine mapping across multiple studies. *bioRxiv.* 2020;doi:10.1101/2020.01.15.908517.
9. Albert A. *Regression and the Moore-Penrose pseudoinverse.* New York, NY: Academic Press; 1972.
10. Maller JB, McVean G, Byrnes J, Vukcevic D, Palin K, Su Z, et al. Bayesian refinement of association signals for 14 loci in 3 common diseases. *Nature Genetics.* 2012;44(12):1294–1301.
11. Brent RP. *Algorithms for minimization without derivatives.* Mineola, NY: Dover; 2002.
12. Kim Y, Carbonetto P, Stephens M, Anitescu M. A fast algorithm for maximum likelihood estimation of mixture proportions using sequential quadratic programming. *Journal of Computational and Graphical Statistics.* 2020;29(2):261–273.
13. Sudlow C, Gallacher J, Allen N, Beral V, Burton P, Danesh J, et al. UK Biobank: an open access resource for identifying the causes of a wide range of complex diseases of middle and old age. *PLoS Medicine.* 2015;12(3):e1001779.
14. Bycroft C, Freeman C, Petkova D, Band G, Elliott LT, Sharp K, et al. The UK Biobank resource with deep phenotyping and genomic data. *Nature.* 2018;562(7726):203–209.
15. Canela-Xandri O, Rawlik K, Tenesa A. An atlas of genetic associations in UK Biobank. *Nature Genetics.* 2018;50(11):1593–1599.

16. Lee Y, Luca F, Pique-Regi R, Wen X. Bayesian multi-SNP genetic association analysis: control of FDR and use of summary statistics. bioRxiv. 2018;doi:10.1101/316471.
17. Wen X, Lee Y, Luca F, Pique-Regi R. Efficient integrative multi-SNP association analysis via deterministic approximation of posteriors. *American Journal of Human Genetics*. 2016;98(6):1114–1129.
18. R: a language and environment for statistical computing; 2018. Available from: <https://www.R-project.org>.
19. Zou Y, Carbonetto P, Wang G, Stephens M. stephenslab/dsc_susiers: release of dsc_susiers repository accompanying publication in PLoS Genetics; 2022. Available from: <https://doi.org/10.5281/zenodo.5611713>.