Stochastic search variable selection for split-plot and blocked screening designs (supplementary material)

To perform the Gibbs sampling, we use WinBUGS (Spiegelhalter et al., 2003) software to construct our code. Table S1 provides a general code for the Gibbs sampling algorithm, which can be used to analyze data from any $n \times m$ SPDS design with $w$ whole plots and $n \times m$ BDS design with $w$ blocks. A list of the variables in the code and their corresponding model notations is given in Table S2. Let $\beta_i$ be the $i$th element in $\beta$, where $i = 0, 1, \cdots, q$; $\delta_i$ be the $i$th element in $\delta$, where $i = 1, \cdots, q$; $\gamma_i$ be the $i$th element in $\gamma$, where $i = 1, \cdots, w$; and $y_i$ be the $i$th element in $y$, where $i = 1, \cdots, n$. Table S1 consists of three main parts. The first part (lines 2–15) sets the probability for $\delta_i = 1$, where variables $p$, $p_0$, $p_1$, $p_{00}$, $p_{01}$, $p_{10}$, and $p_{11}$ in the code correspond to the tuning parameters $p$, $p_0$, $p_1$, $p_{00}$, $p_{01}$, $p_{10}$, and $p_{11}$, respectively; variable $m$ corresponds to the number of factors $m$; variable delta is a $1 \times q$ vector used for storing the values of $\delta_i$’s; and variable pi is a $1 \times q$ vector used for storing the probabilities for $\delta_i = 1$. The second part (lines 16–17) sets variances for $\beta_i$’s, where variables $c$ and $\tau$ correspond to the tuning parameters $c$ and $\tau$, respectively. Variable sig2.b is a $1 \times 2$ vector, where sig2.b[1] stores the variance of the main effect or two-factor interaction $\beta_i$ ($i = 1, \cdots, q - m$) when $\delta_i = 0$, that is, $\tau^2$, and sig2.b[2] stores the variance of $\beta_i$ when $\delta_i = 1$, that is, $c^2\tau^2$. As mentioned in Section 3.1 (also shown in
Table S1: WinBUGS code of Gibbs sampling for three-level or definitive screening designs.

```plaintext
model{
  for(i in 1:m){
    pi[i]<-p
  }
  for(i in 1:(m-1)){
    for(j in (i+1): m){
      pi[m+(i-1)*(m-i)+(i-1)*i/2+(j-i)]<-p00*(1-delta[i])*(1-delta[j])
      +p01*(1-delta[i])*delta[j]
      +p10*delta[i]*(1-delta[j])
      +p11*delta[i]*delta[j]
    }
  }
  for(i in 1:m){
    pi[m+m*(m-1)/2+i]<-p0*(1-delta[i])+p1*delta[i]
  }
  sig2.b[1]<-pow(tau,2)
  sig2.b[2]<-pow(tau,2)*pow(c,2)
  b0~dnorm(0,1.0E-6)
  for(i in 1:(q-m)){
    inv.sig2.b[i]<-1/sig2.b[ind.b[i]]
    b[i]~dnorm(0,inv.sig2.b[i])
  }
  for(i in (q-m+1):q){
    inv.sig2.b[i]<-1/(sig2.b[ind.b[i]]*4)
    b[i]~dnorm(0,inv.sig2.b[i])
  }
  for(i in 1:q){
    delta[i]~dbern(pi[i]); ind.b[i]<-delta[i]+1
  }
  for(i in 1:w){
    gam[i]~dnorm(0,inv.sig2.gam)
  }
  sig.eps~dunif(1,100); inv.sig2.eps<-1/pow(sig.eps,2)
  sig.gam~dunif(1,100); inv.sig2.gam<-1/pow(sig.gam,2)
  for(i in 1:n){
    y[i]~dnorm(mu[i],inv.sig2.eps)
    mu[i]<-b0+inprod(b[],x[i])+gam[wp[i]]
  }
}
```
Table S2: Names of variables for the code in Table S1 and their corresponding model notations.

<table>
<thead>
<tr>
<th>Names of variables</th>
<th>Model notations</th>
<th>Names of variables</th>
<th>Model notations</th>
</tr>
</thead>
<tbody>
<tr>
<td>b[i]</td>
<td>$\beta_i \ (i \neq 0)$</td>
<td>p10</td>
<td>$p_{10}$</td>
</tr>
<tr>
<td>b0</td>
<td>$\beta_0$</td>
<td>p11</td>
<td>$p_{11}$</td>
</tr>
<tr>
<td>c</td>
<td>$c$</td>
<td>pi[i]</td>
<td>(conditional) probability of $\delta_i = 1$</td>
</tr>
<tr>
<td>delta[i]</td>
<td>$\delta_i$</td>
<td>q</td>
<td>q</td>
</tr>
<tr>
<td>gam[i]</td>
<td>$\gamma_i$</td>
<td>sig. eps</td>
<td>$\sigma_\epsilon$</td>
</tr>
<tr>
<td>ind.b[i]</td>
<td>$\delta_i + 1$</td>
<td>sig. gam</td>
<td>$\sigma_\gamma$</td>
</tr>
<tr>
<td>inv.sig2.b[1]</td>
<td>$1/\tau^2$</td>
<td>sig2.b[1]</td>
<td>$\tau^2$</td>
</tr>
<tr>
<td>inv.sig2.eps</td>
<td>$1/\sigma_\epsilon^2$</td>
<td>tau</td>
<td>$\tau$</td>
</tr>
<tr>
<td>inv.sig2.gam</td>
<td>$1/\sigma_\gamma^2$</td>
<td>w</td>
<td>w</td>
</tr>
<tr>
<td>m</td>
<td>$m$</td>
<td>wp[i]</td>
<td>index of the whole plot or block that $y_i$ belongs to</td>
</tr>
<tr>
<td>mu[i]</td>
<td>mean of $y_i$</td>
<td>x[i,]</td>
<td>$i$th row of model matrix (excluding the intercept)</td>
</tr>
<tr>
<td>p</td>
<td>$p$</td>
<td>y[i]</td>
<td>$y_i$</td>
</tr>
<tr>
<td>p0</td>
<td>$p_0$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>p00</td>
<td>$p_{00}$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>p01</td>
<td>$p_{01}$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>p1</td>
<td>$p_1$</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
levels of quadratic effects range from 0 to 1, which are different from those of main effects and two-factor interactions (ranging from \(-1\) to 1). To ensure that all effects are interpreted on the same scale, variances of quadratic effects should be four times those of main and two-factor interactions (Gilmour and Goos, 2009). Thus, the variance of quadratic effect \(\beta_i\) \((i = q - m + 1, \ldots, q)\) is \(4\times\text{sig2.b}[1]\) when \(\delta_i = 0\) and \(4\times\text{sig2.b}[2]\) when \(\delta_i = 1\) (lines 23–26). The third part (lines 18–38) conducts Gibbs sampling, where \(b_0\) corresponds to \(\beta_0\); \(q\) corresponds to \(q\); \(w\) corresponds to the number of whole plots or blocks \(w\); \(b\) is a \(1 \times q\) vector that stores values of \(\beta_1, \ldots, \beta_q\); \(\delta\) is a \(1 \times q\) vector that corresponds to \(\delta'\); \(\gamma\) is a \(1 \times q\) vector that corresponds to \(\gamma'\); \(y\) is a \(1 \times n\) vector that corresponds to \(y'\); and \(x\) is an \(n \times q\) matrix that corresponds to the model matrix \((x[,1], \ldots, x[,m])\) are associated with main effects \(\beta_1, \ldots, \beta_m\); \(x[,m+1], \ldots, x[,q-m]\) are associated with two-factor interactions \(\beta_{12}, \beta_{13}, \ldots, \beta_{1m}, \beta_{23}, \ldots, \beta_{(m-1)m}\); and \(x[,q-m+1], \ldots, x[,q]\) are associated with quadratic effects \(\beta_{11}, \ldots, \beta_{mm}\). The Gibbs sampling algorithm consists of the following processes:

1. a Normal draw for \(\beta_0|\{\beta_j\}_{j \neq 0}, \delta, \gamma, \sigma_\gamma, \sigma_\epsilon\) (line 18).

2. \(q\) Normal draws for \(\beta_i|\{\beta_j\}_{j \neq i}, \delta, \gamma, \sigma_\gamma, \sigma_\epsilon\), where \(i = 1, \ldots, q\) (lines 19–22 indicate the main effects and two-factor interactions, and lines 23–26 indicate the quadratic effects, where \(\text{inv.sigt2.b}\) stores the inverse of \(\text{sig2.b}\); and \(\text{ind.b}[i]\) indicates whether \(\beta_i\) is active \((\text{ind.b}[i] = 2)\) or not \((\text{ind.b}[i] = 1)\)).

3. \(q\) Bernoulli draws for \(\delta_i|\beta, \{\delta_j\}_{j \neq i}, \gamma, \sigma_\gamma, \sigma_\epsilon\), where \(i = 1, \ldots, q\) (lines 27–29).

4. \(w\) Normal draws for \(\gamma_i|\beta, \delta, \{\gamma_j\}_{j \neq i}, \sigma_\gamma, \sigma_\epsilon\), where \(i = 1, \ldots, w\) (lines 30–32, where \(\text{inv.sigt2.gam}\) corresponds to \(1/\sigma_\gamma^2\)).

5. a Uniform draw for \(\sigma_\epsilon|\beta, \delta, \gamma, \sigma_\gamma\) (line 33, where \(\text{sig.eps}\) corresponds to \(\sigma_\epsilon\), and \(\text{inv.sigt2.eps}\) corresponds to \(1/\sigma_\epsilon^2\)).
6. a Uniform draw for $\sigma_\gamma | \beta, \delta, \gamma, \sigma_\varepsilon$ (line 34, where sig.gam corresponds to $\sigma_\gamma$).

7. $n$ Normal draws for $y_i | \beta, \delta, \gamma, \sigma_\gamma, \sigma_\varepsilon$, where $i = 1, \cdots, n$ (lines 35–38, where mu stores means of $y$ and wp[i] indicates which whole plot or block $y_i$ belongs to).

To perform Gibbs sampling using WinBUGS, we have to add sections DATA and INITS. In the DATA section, the following should be provided: $n \times 1$ vector of responses ($y$), $n \times q$ model matrix ($x$, excluding the column for the intercept), $n \times 1$ vector of the whole plot or block indexes ($wp$), number of runs ($n$), number of factors ($m$), number of whole plots or blocks ($w$), number of latent variables ($q$), and tuning parameters ($p$, $p0$, $p1$, $p00$, $p01$, $p10$, $p11$, $c$, and $tau$). In the INITS section, a list of initial values for the following variables should be specified: $b0$, $b$, delta, gam, sig.gam, and sig.eps. In the event the screening design is a two-level design rather than a definitive screening design, quadratic effects can be dropped from the winBUGS code (see Table S3).

References


Table S3: WinBUGS code of Gibbs sampling for two-level screening designs.

```plaintext
model{
  for(i in 1:m){
    pi[i]<-p
  }
  for(i in 1:(m-1)){
    for(j in (i+1): m){
      pi[m+(i-1)*(m-i)+(i-1)*i/2+(j-i)]<-p00*(1-delta[i])*(1-delta[j])
      +p01*(1-delta[i])*delta[j]
      +p10*delta[i]*(1-delta[j])
      +p11*delta[i]*delta[j]
    }
  }
  sig2.b[1]<-pow(tau,2)
  sig2.b[2]<-pow(tau,2)*pow(c,2)
  b0~dnorm(0,1.0E-6)
  for(i in 1:q){
    inv.sig2.b[i]<-1/sig2.b[ind.b[i]]
    b[i]~dnorm(0,inv.sig2.b[i])
  }
  for(i in 1:w){
    delta[i]~dbern(pi[i]); ind.b[i]<-delta[i]+1
  }
  for(i in 1:w){
    gam[i]~dnorm(0,inv.sig2.gam)
  }
  sig.eps~dunif(1,100); inv.sig2.eps<-1/pow(sig.eps,2)
  sig.gam~dunif(1,100); inv.sig2.gam<-1/pow(sig.gam,2)
  for(i in 1:n){
    y[i]~dnorm(mu[i],inv.sig2.eps)
    mu[i]<-b0+inprod(b[],x[i,])+gam[wp[i]]
  }
}
```