Biostatistics (2017), **0**, 0, *pp.* 1–17
doi:10.1093/biostatistics/DualNHoodBiosSuppAppend

Supplementary Materials: Modeling a Bivariate Residential-Workplace Neighborhood Effect when Estimating the Effect of Proximity to Fast-Food Establishments on Body Mass Index

A. JAMES O'MALLEY*, PETER JAMES, TODD A. MACKENZIE, JINYOUNG BYUN,

SUBU V. SUBRAMANIAN, JASON P. BLOCK

Department of Biomedical Data Science and The Dartmouth Institute of Health Policy and Clinical Practice, 1 Medical Center Drive, Geisel School of Medicine at Dartmouth, Lebanon, NH 03755, USA

James.OMalley@Dartmouth.edu, Todd.MacKenzie@Dartmouth.edu,

Jinyoung.Byun@Dartmouth.edu

Departments of Epidemiology and Environmental Health at the Harvard TH Chan School of Public Health, Boston, MA 02115, USA

pjames@hsph.harvard.edu, svsubram@hsph.harvard.edu

Department of Population Medicine at the Harvard Pilgrim Health Care Institute and Harvard Medical School, Boston, MA 02115, USA

jblock1@partners.org

*To whom correspondence should be addressed.

1. Self-developed MCMC Procedure: Conditional posterior distributions

Although we were able to estimate the model in JAGS (and WinBUGS), deriving the conditional posterior distributions provides important insights into the manner by which the data informs the model parameters. They also allow the calculations performed in JAGS to be checked and the computational time for the simulation to be dramatically reduced. The following describes our hybrid Gibbs sampler (Gelfand and Smith, 1990; Geman and Geman, 1984) – Metropolis-Hastings (Chib and Greenberg, 1995; Metropolis *and others*, 1953; Hastings, 1970) Markov-chain Monte-Carlo (MCMC) algorithm.

1.1 *Overview and novel contribution*

The prior distributions for $\mathbf{\Sigma}_{\text{ind}}$ and $\mathbf{\Sigma}_{\text{area}}$ have conjugate forms in all analyses, allowing the conditional posterior distribution of every parameter other than ρ_{e} to be evaluated in closed form. The marginal distribution for ϵ_{i1} combined with the AR(1) structure on ϵ_{it}, $t = 2, \ldots, T = 8$, inhibits closed-form determination of the conditional posterior for ρ_{e}. However, we derive a close approximation to the conditional posterior distribution for ρ_{e} by substituting ρ_{e} with its conditional method-of-moments estimator in the expression for the variance and then use the resulting closed-form approximation to accept or reject the randomly generated candidate values. The resulting independence sampler (Tierney, 1994) draws candidate values of ρ_{e} with high Metropolis-Hastings (M-H) acceptance probabilities. For the analysis of the proximity to food data, the M-H acceptance rate for ρ_{e} under (1.1) exceeds 90% implying that new parameter values are drawn almost as frequently as for a Gibbs (sampler) step. Unlike Gibbs steps, which are random walks and thus yield serially correlated draws, the accept-reject MCMC step yields new values of ρ_{e} that are independent of prior values, improving the efficiency of the computation.

1.2 *Notation and Set-up*

Recall that Y_{it}, $\boldsymbol{X}_{it}$, R_{it}, and W_{it} denote the outcome (BMI), a vector of fixed-effect predictors, neighborhood of residence, and neighborhood of workplace for the ith ($i = 1, \ldots, N$) individual at their t'th medical exam (measurement occasion referred to as "exam", $t = 1, \ldots, T = 8$). The general form of the model for the proximity to food analysis is given by:

$$Y_{it} = \boldsymbol{\beta}^T \boldsymbol{X}_{it} + \mu_{i0} + \mu_{i1} t + \theta_{R_{it}} + \gamma_{W_{it}} + \epsilon_{it}, \tag{1.1}$$

where $\boldsymbol{\mu}_i = (\mu_{i0}, \mu_{i1})^T \sim \text{Normal}_2(0, \boldsymbol{\Sigma}_{\text{ind}})$, $\boldsymbol{\eta}_k = (\theta_k, \gamma_k)^T \sim \text{Normal}_2(0, \boldsymbol{\Sigma}_{\text{area}})$ for $k = 1, \ldots, K$ (K distinct areas), and $\epsilon_{it} \mid \epsilon_{i(t-1)} \sim \text{Normal}(\rho_{\text{e}} \epsilon_{i(t-1)}, \sigma^2_{\text{e}}(1 - \rho^2_{\text{e}}))$ for all i, t. The distribution specified for $\epsilon_{it} \mid \epsilon_{i(t-1)}$ is consistent with the marginal distribution $\epsilon_{it} \sim N(0, \sigma^2_{\text{e}})$. Unless otherwise stated, the following prior distributions were assumed: $p(\boldsymbol{\beta}) \propto 1$, $\boldsymbol{\Sigma}_{\text{ind}} \sim IW(a, \boldsymbol{A})$, $\boldsymbol{\Sigma}_{\text{area}} \sim IW(b, \boldsymbol{B})$, $\sigma^2_{\text{e}} \sim IG(\nu_1, \nu_2)$, and $\rho_{\text{e}} \sim U(-1, 1)$.

Let $S_i \subset \{1, \ldots, T = 8\}$ for $i = 1, \ldots, N$ denote the set of exams individual i attended, $S_r = \{i : \sum_{t=1}^{T} I(R_{it} = r) > 0\}$ for $r = 1, \ldots, K$ denote the set of individuals who reside in neighborhood r at some point, and $S_w = \{i : \sum_{t=1}^{T} I(W_{it} = w) > 0\}$ for $w = 1, \ldots, K$ denote the set of individuals who work in neighborhood w at some point. The cardinality of S_i, S_r, and S_w are denoted n_i, n_r, and n_w, respectively.

Then let $\boldsymbol{X}_i$ denote the $n_i \times p$, $p = \dim(\boldsymbol{X}_{it})$, matrix of predictors for the observations on individual i (the k'th row of $\boldsymbol{X}_i$ contains the observations from individual i's k'th exam, which differs from study wave k if they missed a prior exam), $\boldsymbol{Z}_{0i} = (\boldsymbol{j}_i, \boldsymbol{e}_i)$ be the $n_i \times 2$ matrix in which $\boldsymbol{j}_i$ is a vector of ones of length n_i and $\boldsymbol{e}_i$ is the vector counterpart of S_i. Also let $\boldsymbol{Z}_{1i}$ and $\boldsymbol{Z}_{2i}$ be the n_i by K matrices containing 1 in element (t, k) if $R_{it} = k$ and $W_{it} = k$, respectively. The vectorized form of the model is then

$$\boldsymbol{Y}_i = \boldsymbol{X}_i \boldsymbol{\beta} + \boldsymbol{Z}_{0i} \boldsymbol{\mu}_i + \boldsymbol{Z}_{1i} \theta + \boldsymbol{Z}_{2i} \gamma + \boldsymbol{\epsilon}_i, \tag{1.2}$$

where $\boldsymbol{\epsilon}_i \sim \text{Normal}(0, \sigma^2_{\text{e}} \boldsymbol{V}_i(\rho_e))$, $V(\rho_e)_{hj} = \rho_e^{|h-j|}$ and $\boldsymbol{V}_i = \boldsymbol{V}_i(\rho_e)$ is the sub-matrix of $\boldsymbol{V}(\rho_e)$

corresponding to S_i.

For separate updates of θ_k and γ_k, the conditional priors $p(\theta_k \mid \gamma_k)$ and $p(\gamma_k \mid \theta_k)$ are needed. It follows from results for normal distributions that $\theta_k \mid \gamma_k \sim N(\lambda_{\theta_k|\gamma_k}, \tau^2_{\theta_k|\gamma_k})$ and $\gamma_k \mid \theta_k \sim N(\lambda_{\gamma_k|\theta_k}, \tau^2_{\gamma_k|\theta_k})$, where $\lambda_{\theta_k|\gamma_k} = (\rho_{\text{area}}\sigma_{\text{home}}/\sigma_{\text{work}})\gamma_k$, $\tau^2_{\theta_k|\gamma_k} = \sigma^2_{\text{home}} - \rho_{\text{area}}\sigma_{\text{home}}/\sigma_{\text{work}}$, $\lambda_{\gamma_k|\theta_k} = (\rho_{\text{area}}\sigma_{\text{work}}/\sigma^2_{\text{home}})\theta_k$, and $\tau^2_{\gamma_k|\theta_k} = \sigma^2_{\text{work}} - \rho_{\text{area}}\sigma_{\text{work}}/\sigma_{\text{home}}$.

1.3 *Conditional Posterior of $\boldsymbol{\beta}$*

Let $\tilde{\boldsymbol{Y}}_i^{(1)} = \boldsymbol{Y}_i - \boldsymbol{Z}_{0i}\boldsymbol{\mu}_i - \boldsymbol{Z}_{1i}\boldsymbol{\theta} - \boldsymbol{Z}_{2i}\boldsymbol{\gamma}$. Under the flat non-informative prior specified above, the conditional posterior of $\boldsymbol{\beta}$ given the data $\boldsymbol{D} = (\boldsymbol{Y}, \boldsymbol{X}, \boldsymbol{R}, \boldsymbol{W})_{i=1:N,t=1:8}$ and the other parameters is

$$(\boldsymbol{\beta} \mid .) \sim \text{Normal}_p\left(\hat{\boldsymbol{\beta}}, (\sum_{i=1}^{N} \boldsymbol{X}_i^T \boldsymbol{V}_i^{-1} \boldsymbol{X}_i)^{-1}\right),$$

where $\hat{\boldsymbol{\beta}} = (\sum_{i=1}^N \boldsymbol{X}_i^T \boldsymbol{V}_i^{-1} \boldsymbol{X}_i)^{-1} \sum_{i=1}^N \boldsymbol{X}_i^T \boldsymbol{V}_i^{-1} \tilde{\boldsymbol{Y}}_i^{(1)}$ is the weighted least squares estimator.

1.4 *Conditional Posterior of $\boldsymbol{\mu}_i$*

Let $\tilde{\boldsymbol{Y}}_i^{(2)} = \boldsymbol{Y}_i - \boldsymbol{X}_i\boldsymbol{\beta} - \boldsymbol{Z}_{1i}\boldsymbol{\theta} - \boldsymbol{Z}_{2i}\boldsymbol{\gamma}$. Then the conditional posterior of $\boldsymbol{\mu}_i$ is

$$(\boldsymbol{\mu}_i \mid .) \sim \text{Normal}_2\left((\boldsymbol{Z}_{0i}^T \boldsymbol{V}_i^{-1} \boldsymbol{Z}_{0i} + \boldsymbol{\Sigma}_{\text{ind}}^{-1})^{-1} \boldsymbol{Z}_{0i}^T \boldsymbol{V}_i^{-1} \tilde{\boldsymbol{Y}}_i^{(2)}, (\boldsymbol{Z}_{0i}^T \boldsymbol{V}_i^{-1} \boldsymbol{Z}_{0i} + \boldsymbol{\Sigma}_{\text{ind}}^{-1})^{-1}\right).$$

1.5 *Conditional Posterior of θ_r*

Let $\tilde{\boldsymbol{Y}}_i^{(3)} = \boldsymbol{Y}_i - \boldsymbol{X}_i\boldsymbol{\beta} - \boldsymbol{Z}_{0i}\boldsymbol{\mu}_i - \boldsymbol{Z}_{2i}\boldsymbol{\gamma}$. In addition, let $\boldsymbol{j}_{ir}$ and $\boldsymbol{V}_{ir}$ be the vector of ones and the sub-matrix of $\boldsymbol{V}(\rho)$ corresponding to the elements of $S_{ir} = \{t : R_{it} = r\}$, respectively. Then the conditional posterior of θ_r given all observed data and all other parameters (including γ_r) is:

$$(\theta_r \mid .) = \text{Normal}_2\left(\hat{\theta}_r, (\sum_{i \in S_r} \boldsymbol{j}_{ir}^T \boldsymbol{V}_{ir}^{-1} \boldsymbol{j}_{ir} + \tau^{-2}_{\theta_r|\gamma_r})^{-1}\right),$$

where $S_r = \{i : \sum_{t=1}^{T} I(R_{it} = r) > 0\}$ and $\hat{\theta}_r = (\sum_{i \in S_r} \boldsymbol{j}_{ir}^T \boldsymbol{V}_{ir}^{-1} \boldsymbol{j}_{ir} + \tau_{\theta_r|\gamma_r}^{-2})^{-1} (\sum_{i \in S_r} \boldsymbol{j}_{ir}^T \boldsymbol{V}_{ir}^{-1} \tilde{\boldsymbol{Y}}_i^{(3)} + \tau_{\theta_r|\gamma_r}^{-2} \lambda_{\theta_r|\gamma_r})$. Thus, $\hat{\theta}_r$ is a weighted average of the adjusted outcomes attributed to neighborhood r based on residence.

1.6 *Conditional Posterior of* γ_w

The conditional posterior of γ_w is a mirror image of that for θ_r. Let $\tilde{\boldsymbol{Y}}_i^{(4)} = \boldsymbol{Y}_i - \boldsymbol{X}_i \boldsymbol{\beta} - \boldsymbol{Z}_{0i} \boldsymbol{\mu}_i - \boldsymbol{Z}_{1i} \boldsymbol{\theta}$. In addition, let $\boldsymbol{j}_{iw}$ and $\boldsymbol{V}_{iw}$ be the vector of ones and the sub-matrix of $\boldsymbol{V}(\rho)$ corresponding to the elements of $S_{iw} = \{t : W_{it} = w\}$. Then the conditional posterior of γ_w given all observed data and other parameters (including θ_w) is:

$$(\gamma_w \mid .) = \text{Normal}_2 \left(\hat{\gamma}_w, (\sum_{i \in S_w} \boldsymbol{j}_{iw}^T \boldsymbol{V}_{iw}^{-1} \boldsymbol{j}_{iw} + \tau_{\gamma_w|\theta_w}^{-2})^{-1} \right),$$

where $S_w = \{i : \sum_{t=1}^{T} I(W_{it} = w) > 0\}$ and $\hat{\gamma}_w = (\sum_{i \in S_w} (\boldsymbol{j}_{iw}^T \boldsymbol{V}_{iw}^{-1} \boldsymbol{j}_{iw} + \tau_{\gamma_w|\theta_w}^{-2})^{-1} (\sum_{i \in S_w} \boldsymbol{j}_{iw}^T \boldsymbol{V}_{iw}^{-1} \tilde{\boldsymbol{Y}}_i^{(4)} + \tau_{\gamma_w|\theta_w}^{-2} \lambda_{\gamma_w|\theta_w})$. Thus, $\hat{\gamma}_w$ is a weighted average of the adjusted outcomes attributed to neighborhood w based on workplace.

1.7 *Novel step: Conditional Posterior of* ρ_{e}

The serial correlation ρ_{e} is a measure of dependence between the residuals, expressed in vectorized form as $\boldsymbol{\epsilon}_i = \boldsymbol{Y}_i - \boldsymbol{X}_i \beta - \boldsymbol{Z}_{0i} \boldsymbol{\mu}_i - \boldsymbol{Z}_{1i} \boldsymbol{\theta} - \boldsymbol{Z}_{2i} \boldsymbol{\gamma}$. In the case of a uniform $(-1, 1)$ prior for ρ_{e}, the conditional posterior distribution of ρ_{e} is proportional to the density of the joint distribution of $\boldsymbol{\epsilon}_i$ truncated to the $(-1, 1)$ interval. Under the AR(1) model, the joint distribution of $\boldsymbol{\epsilon}_i$ reduces to the product of the densities for ϵ_{i1}, $(\epsilon_{i2} \mid \epsilon_{i1})$, ..., $(\epsilon_{iT} \mid \epsilon_{i(T-1)})$, illustrating that ρ_{e} is informed by ϵ_{i1} in addition to the sample correlation between ϵ_{it} and $\epsilon_{i(t-1)}$ for $t > 1$. Because the marginal variance of the first residual is given by $\text{var}(\epsilon_{i1}) = \sigma_{\text{e}}^2$, the conditional posterior of

ρ_{e} is proportional to:

$$p(\rho_{\mathrm{e}} \mid .) \propto \text{Normal}\left(\rho_{\mathrm{e}}; \hat{\rho}_{\mathrm{e}}\left(\frac{\sum_{i,t=2:T}\epsilon_{it}^2}{\sum_{i,t=2:T}\epsilon_{i(t-1)}^2}\right)^{1/2}, \frac{\sigma_{\mathrm{e}}^2(1-\rho_e^2)}{\sum_{i,t=2:T}\epsilon_{i(t-1)}^2}\right) I(-1,1),$$

where $\hat{\rho}_{\mathrm{e}} = \sum_{i,t=2:T}\epsilon_{i(t-1)}\epsilon_{it}/(\sum_{i,t=2:T}\epsilon_{i(t-1)}^2 \sum_{i,t=2:T}\epsilon_{it}^2)^{1/2}$ is the Pearson correlation coefficient evaluated on $\{\epsilon_{i(t-1)}, \epsilon_{it}\}_{i=1,\ldots,N;t=2,\ldots,T}$.

Because $p(\rho_{\mathrm{e}} \mid .)$ is not available in closed-form, to draw values of ρ_{e} we use the Metropolis-Hastings algorithm with the candidate generating distribution based on the approximation of the true conditional posterior for $p(\rho_{\mathrm{e}} \mid .)$ given in step 3 below. Let $\rho_{\mathrm{e,old}}$ denote the current value of ρ_{e}. The M-H step is then:

1. Generate the candidate value $\rho_{\mathrm{e,new}} \sim \text{Normal}\left(\left(\frac{\sum_{i,t=2:T}\epsilon_{it}^2}{\sum_{i,t=2:T}\epsilon_{i(t-1)}^2}\right)^{1/2}\hat{\rho}_{\mathrm{e}}, \frac{\sigma_{\mathrm{e}}^2(1-\hat{\rho}_e^2)}{\sum_{i,t=2:T}\epsilon_{i(t-1)}^2}\right)$

2. Repeat step 1 until $\rho_{\mathrm{e,new}} \in (-1,1)$ is obtained.

3. Evaluate the acceptance probability

$$q = \left(\frac{1-\rho_{\mathrm{e,old}}^2}{1-\rho_{\mathrm{e,new}}^2}\right)^{1/2} \exp\left[-\left(\frac{1}{1-\rho_{\mathrm{e,new}}} - \frac{1}{1-\hat{\rho}_{\mathrm{e}}}\right)\left(\rho_{\mathrm{e,new}} - \left(\frac{\sum_{i,t=2:T}\epsilon_{it}^2}{\sum_{i,t=2:T}\epsilon_{i(t-1)}^2}\right)^{1/2}\hat{\rho}_{\mathrm{e}}\right)^2 + \left(\frac{1}{1-\rho_{\mathrm{e,old}}} - \frac{1}{1-\hat{\rho}_{\mathrm{e}}}\right)\left(\rho_{\mathrm{e,old}} - \left(\frac{\sum_{i,t=2:T}\epsilon_{it}^2}{\sum_{i,t=2:T}\epsilon_{i(t-1)}^2}\right)^{1/2}\hat{\rho}_{\mathrm{e}}\right)^2\right]$$

4. If $q \geqslant 1$ accept $\rho_{\mathrm{e,new}}$ as the new value of ρ_{e}. If $q < 1$ generate $u \sim U(0,1)$ and accept $\rho_{\mathrm{e,new}}$ as the new value of ρ_{e} if $u < q$; otherwise, retain $\rho_{\mathrm{e,old}}$ as the value of ρ_{e}.

We obtain values of ϵ_{it} by substituting the current values of the model parameters to evaluate the vector $\boldsymbol{\epsilon}_i = \boldsymbol{Y}_i - \boldsymbol{X}_i\boldsymbol{\beta} - \boldsymbol{Z}_{0i}\boldsymbol{\mu}_i - \boldsymbol{Z}_{1i}\boldsymbol{\theta} - \boldsymbol{Z}_{2i}\boldsymbol{\gamma}$.

The rationale for the above procedure is that the true posterior distribution is closely approximated by the posterior generated from the Pearson correlation coefficient evaluated on $\{\epsilon_{i(t-1)}, \epsilon_{it}\}_{i=1,\ldots,N;t=2,\ldots,T}$. The quantity $(\sum_{i,t=2:T}\epsilon_{i(t-1)}^2)/(\sum_{i,t=1:(T-1)}\epsilon_{it}^2)$ should be very close to 1 as the numerator and denominator are summing over many of the same terms. Therefore,

the candidate generating density for ρ_{e} is a distribution whose mean is close to $\hat{\rho}_{\mathrm{e}}$, the Pearson correlation coefficient. Therefore, we can use a Metropolis-Hastings independence sampler with candidate values of ρ_{e} drawn from the approximate posterior distribution implied by steps 1 and 2 of the above accept-reject algorithm (Berger *and others*, 2005). We anticipate that the acceptance probability, q, will be close to 1 in practical applications such as those considered in this paper.

1.8 *Conditional Posterior of* σ_{e}^2

As for the conditional posterior of ρ_{e}, to determine the conditional posterior of σ_{e}^2 we use the fact that the joint distribution of $\boldsymbol{\epsilon}_i$ expands into the product of the marginal distribution of ϵ_{i1} and the conditional densities of $(\epsilon_{it} \mid \epsilon_{i(t-1)})$ for $i = 2, \ldots, T$. Let $\delta_{it} = \epsilon_{it} - \rho_{\mathrm{e}}\epsilon_{i(t-1)}$ and $N_{\mathrm{tot}} = \sum_{i=1} n_i$. Then

$$(\sigma_{\mathrm{e}}^2 \mid .) = IG\left(\nu_1 + \frac{N_{\mathrm{tot}}}{2}, \left(\frac{1}{\nu_2} + \frac{1}{2}\sum_i \epsilon_{i1}^2 + \frac{1}{2(1-\rho_{\mathrm{e}}^2)}\sum_{i,t=2}\delta_{it}^2\right)^{-1}\right).$$

The middle term in the evaluation of the second parameter of the above inverse-gamma distribution obtains from the first residual for each subject whereas the final term aggregates squared differences of the current residual and its conditional mean given the prior residual (δ_{it}^2). These both contribute information about σ_{e}^2.

1.9 *Conditional Posterior of* $\boldsymbol{\Sigma}_{\mathrm{ind}}$

The latent variables (or level II data) that inform the conditional posterior of $\boldsymbol{\Sigma}_{\mathrm{ind}}$ are the bivariate random effects $\boldsymbol{\mu}_i$, $i = 1, \ldots, N$. Because the inverse-Wishart is the conjugate prior, the conditional posterior distribution is given by:

$$(\boldsymbol{\Sigma}_{\mathrm{ind}} \mid .) = IW(a + N, \boldsymbol{A} + \sum_{i=1}^{N} \boldsymbol{\mu}_i \boldsymbol{\mu}_i^T).$$

1.10 *Conditional Posterior of* $\boldsymbol{\Sigma}_{\text{area}}$

The latent variables that inform the conditional posterior of $\boldsymbol{\Sigma}_{\text{area}}$ are the bivariate random effects $\boldsymbol{\eta}_k = (\theta_k, \gamma_k)$, $k = 1, \ldots, K$. Because the inverse-Wishart is the conjugate prior, the conditional posterior distribution is given by:

$$(\boldsymbol{\Sigma}_{\text{area}} \mid .) = IW(b + K, \boldsymbol{B} + \sum_{k=1}^{K} \boldsymbol{\eta}_k \boldsymbol{\eta}_k^T).$$

1.11 *Conditional Posterior under Product-Normal Priors*

Under the product-normal prior $\mu_{i0} \sim N(0, \tau_0^2)$, $\mu_{i1} \mid \mu_{i0} \sim N(\phi\mu_{i0}, \tau_1^2)$, $\phi \sim N(0, v_\phi^2)$ and $\tau_j^2 \sim \text{IG}(\omega_1, \omega_2)$, $j = 0, 1$. The conditional posterior for $\boldsymbol{\mu}_i = (\mu_{i0}, \mu_{i1})$ is structurally unchanged from the inverse-Wishart prior specification for $\boldsymbol{\Sigma}_{\text{ind}}$. It follows that

$$(\tau_0^2 \mid .) = \text{IG}\left(\omega_1 + \frac{N}{2}, \left(\frac{1}{\omega_2} + \frac{1}{2}\sum_{i=1}^{N} \mu_{0i}^2\right)^{-1}\right),$$

$$(\tau_1^2 \mid .) = \text{IG}\left(\omega_1 + \frac{N}{2}, \left(\frac{1}{\omega_2} + \frac{1}{2}\sum_{i=1}^{N} (\mu_{1i} - \phi\mu_{i0})^2\right)^{-1}\right),$$

and

$$(\phi \mid .) = \text{Normal}\left(\frac{\sum_{i=1}^{N} \mu_{i0}\mu_{i1}}{\sum_{i=1}^{N} \mu_{i0}^2 + \tau_1^2/v_\phi^2}, \frac{\tau_1^2}{\sum_{i=1}^{N} \mu_{i0}^2 + \tau_1^2/v_\phi^2}\right).$$

Notably, as $v_\phi^2 \to \infty$ the conditional prior mean of ϕ converges to the ratio of the covariance between the elements of $\mu_1 = (\mu_{11}, \ldots, \mu_{N1})$ and $\mu_0 = (\mu_{10}, \ldots, \mu_{N0})$ and the variance of the elements of μ_0. The elements of $\boldsymbol{\Sigma}_{\text{ind}}$ are then determined via $\sigma_{\text{int}}^2 = \tau_0^2$, $\sigma_{\text{slope}}^2 = \psi^2\tau_0^2 + \tau_1^2$, and $\text{cov}(\mu_{i0}, \mu_{i1}) = \rho_{\text{ind}}\sigma_{\text{int}}\sigma_{\text{slope}} = \psi\tau_0^2$.

Under a product-normal prior for $\boldsymbol{\eta}$, the conditional posterior distributions for the elements of $\boldsymbol{\Sigma}_{\text{area}}$ are analogous to those for $\boldsymbol{\Sigma}_{\text{ind}}$.

2. Convergence Diagnostics for MCMC

The convergence of the MCMC procedure used for Bayesian estimation was examined using CODA (Convergence Diagnostics and Output Delivery) software (Best *and others*, 1995). The behavior of the chain was monitored using trace plots (Hellmich *and others*, 1998) of the sequence of draws for each parameter for different starting values and random number seeds. The corrected scale reduction factor (Brooks and Gelman, 1998) indicated that a burn-in time of 10,000 iterations was appropriate (the 0.975 quantiles were generally less than 1.2).

The half-width test of Heidelberger and Welch (1983) applied with a test accuracy of 0.1 to a chain run for 50,000 iterations past burn-in was satisfied for all parameters. Therefore, 50,000 was a sufficient run-length to obtain posterior inferences to the required level of precision for the model parameters and derived quantities. To err on the side of caution, following a burn-in phase of 20,000 iterations we based inferences on 150,000 draws obtained from 50,000 draws on three independent chains. See Figure 1 shows the trace plot from three different starting points and random number seeds of the posterior mean for ρ_{area} in the primary analysis; the numerically very similar results confirm convergence and that the Monte-Carlo standard error had largely dissipated.

References

Berger, J. O., Strawderman, W. and Tang, D. (2005). Posterior propriety and admissibility of hyperpriors in normal hierarchcial models. *The Annals of Statistics* **33**, 606–646.

Best, N., Cowles, M. K. and Vines, K. (1995). *Convergence Diagnosis and Output Analysis Software for Gibbs Sampling Output.*. MRC Biostatistics Unit, Institute of Public Health, Robinson Way, Cambridge CB2 2SR, UK.

Brooks, Stephen P. and Gelman, Andrew. (1998). General methods for monitoring conver-

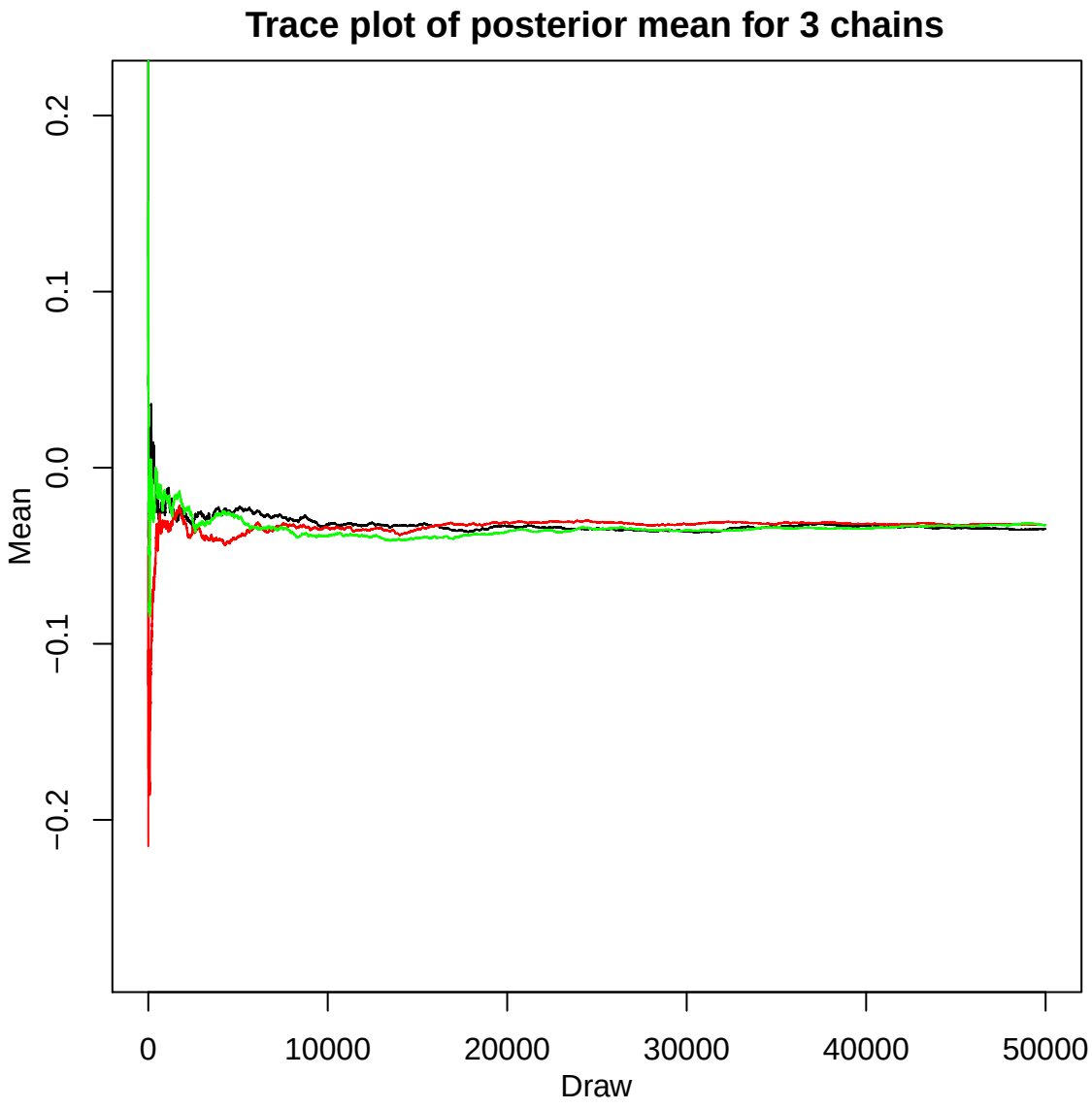

Fig. 1. Trace plot of ρ_{area} for the primary analysis

gence of iterative simulations. *Journal of Computational and Graphical Statistics* **7**, 434–455.

Chib, S. and Greenberg, E. (1995). Understanding the metropolis-hastings algorithm. *The American Statistician* **49**, 327–335.

Gelfand, Alan E. and Smith, Adrian F. M. (1990). Sampling-based approaches to calculating marginal densities. *Journal of the American Statistical Association* **85**, 398–409.

Geman, Stuart and Geman, Donald. (1984). Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **6**, 721–741.

Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications. *Biometrika* **57**, 97–109.

Heidelberger, Philip and Welch, Peter D. (1983). Simulation run length control in the presence of an initial transient. *Operations Research* **31**, 1109–1144.

Hellmich, M., Abrams, K. R., Jones, D. R. and Lambert, P. C. (1998). A Bayesian approach to a general regression model for ROC curves. *Academic Radiology* **18**, 436–443.

Metropolis, N, Rosenbluth, AW, Rosenbluth, MN, Teller, AH and Teller, E. (1953). Equations of state calculations by fast computing machines. *Journal of Chemical Physics* **21**, 1078–92.

Tierney, L. (1994). Markov chains for exploring posterior distributions. *Annals of Statistics* **22**, 1701–1762.

3. Study data

Data from the Framingham Heart Study is not freely available due to patient confidentiality. The Framingham Heart Study has a formal process for requesting data, and anyone can request data

directly via: https://www.framinghamheartstudy.org/researchers/application-review.php. In lieu of the actual data, we provide a pseudo data set named simdata.txt. The size and structure of these data are identical to the actual data, allowing the same code to be used to replicate the Bayesian analysis that yielded the results reported in the main text. The pseudo data, data description, and code are available on GitHub at https://github.com/kiwijomalley/ProximityToFood. For readers' convenience, the data description and code are also provided below in PDF format.

3.1 *Data dictionary*

The data is described in the following:

- ID: The identification number of the individual. Ranges from 1 to 2,889.
- wave: The Framingham Heart Study exam wave; ranges from 1 to 8 for the Offspring cohort and 6 to 8 for the Omni cohort, which only started in wave 6 of the offspring cohort
- Tobs: The total number of exams the individual attended
- neighborhoodHome: The neighborhood where the individual lived
- neighborhoodWork: The neighborhood where an individual was employed (if employed)
- BMI: Body Mass Index
- yob: Year of birth of individual
- smokes: Whether individual is a current smoker
- male: Whether individual has male gender
- married: Whether individual is currently married
- educ: Educational level of individual (0 = High-school or less, 1 = Completed high school; 2 = Other)

- tractpov: Percent of households below the poverty line in neighborhood where individual lived
- etractpov: Percent of households below the poverty line in neighborhood where individual was employed
- unemploy: Whether or not individual was unemployed
- DistHome: Distance in kilometers from individual's home to the nearest fast food establishment
- DistWork: Distance in kilometers from individual's workplace to the nearest fast food establishment (if employed)
- DriveDist: Number of fast food restaurants within a 60 meter buffer of the shortest commute between work and home

The first row of the data set contains the above variable names. Workplace distance, driving distance and the poverty of an individual's workplace neighborhood are only available for employed individuals. For simplicity, real numerical values are given for these variables on such cases. However, the JAGS script is coded so that only observations in which the individual is employed directly contribute to the estimation of model parameters involving these variables.

4. Code

The code for performing the analysis is shown below. The first subsection contains the R code in which the commands to call JAGS are specified, the call to JAGS, and finally post-estimation analysis and presentation of the results. The second subsection shows the JAGS code for the primary analysis. The entire analysis is performed by running the R code as JAGS is called from the R script.

4.1 *R Code*

```
################################################################################################
# Runs analysis in R using RJAGS
# Must first install JAGS and then install RJAGS package from CRAN website
################################################################################################

## R wrapper code to estimate distance-to-food model ##

#Set-up: Load in libraries
library(rjags)
coda.options(combine.plots=TRUE,combine.stats=TRUE)

#Specify filename of JAGS code for Bayesian analysis and data in and output directories
rsource="Enter directory where code is stored" #If run on a Linux server
setwd(rsource)
datdir="../Data/" #Sub-directory to store data
outdir="../Output/" #Sub-directory to store output

#Specify code and initial values to use
JAGScode<-"RealBMICodeIntIW.bug"
IC<-list(be=c(30, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0), isigma=0.25, u=0.5) #Initial values

#Read in data
BMIdata<-read.table(paste(datdir,"simdata.txt",sep=""),sep=" ",col.names=c("ID", "wave", "Omni", "Tobs", "TractHome", "Tract
 "BMI", "yob", "smokes", "male", "married", "educ", "tractpov", "etractpov", "unemploy", "DistHome", "DistWork", "DriveDist"

#Define parameters and operating characteristics of MCMC procedure
parameters = c("be", "sigma", "rho", "taurand", "tauarea", "corrand", "corarea")    # The parameter(s) to be monitored.
adaptSteps = 500              # Number of steps to "tune" the samplers.
burnInSteps = 20000           # Number of steps to "burn-in" the samplers.
nChains = 3                   # Number of chains to run.
numSavedSteps=150000          # Total number of steps in chains to save.
thinSteps=1                   # Number of steps to "thin" (1=keep every step).
nIter = ceiling( ( numSavedSteps * thinSteps ) / nChains ) # Steps per chain.

#Make an object for fitting model using JAGS
jagsModel = model<-jags.model(JAGScode,data=BMIdata,inits=IC,n.chains=nChains)

# Burn-in phase of model estimation:
cat( "Burning in the MCMC chain...\n" )
update( jagsModel , n.iter=burnInSteps )

# Main phase of MCMC model: draws from the posterior distribution are saved in codaSamples
cat( "Sampling final MCMC chain...\n" )
codaSamples = coda.samples( jagsModel , variable.names=parameters ,
                            n.iter=nIter , thin=thinSteps )
# resulting codaSamples object has these indices:
#   codaSamples[[ chainIdx ]][ stepIdx , paramIdx ]

mcmcChain = as.matrix( codaSamples )
summary(mcmcChain)


## Prepare for and implement post model-fitting computations to check whether chain converged ##

beSample0 = mcmcChain[,"be[1]"] # Put sampled values in a vector.
beSample1 = mcmcChain[,"be[2]"] # Put sampled values in a vector.
sigmaSample = mcmcChain[,"sigma"] # Put sampled values in a vector.
```

```
rhoSample = mcmcChain[,"rho"] # Put sampled values in a vector.
tau1Sample = mcmcChain[,"tauarea[1,1]"] # Put sampled values in a vector.
corrandSample = mcmcChain[,"corrand"] # Put sampled values in a vector.
corareaSample = mcmcChain[,"corarea"] # Put sampled values in a vector.

#Plots of sequences of draws
par(mfrow=c(3,2), srt=0, mai=c(0.6, 0.6, 0.4, 0.2), mgp=c(2,1,0))
plot(beSample0[1:nIter],type="l",main="MCMC Chain for Intercept",xlab="iteration",ylab="estimate")
plot(beSample1[1:nIter],type="l",main="MCMC Chain for Wave 2",xlab="iteration",ylab="estimate")
plot(sigmaSample[1:nIter],type="l",main="MCMC Chain for error variance",xlab="iteration",ylab="estimate")
plot(tau1Sample[1:nIter],type="l",main="MCMC Chain for home variance",xlab="iteration",ylab="estimate")
plot(corrandSample[1:nIter],type="l",main="MCMC Chain for individual RE correlation",xlab="iteration",ylab="estimate")
plot(corareaSample[1:nIter],type="l",main="MCMC Chain for area RE correlation",xlab="iteration",ylab="estimate")
dev.copy2eps(file=paste(outdir,"TraceSeqIW4sim.eps",sep=""), width=6, height=6, horizontal=FALSE) #to file

#Trace plots of long-run averages
par(mfrow=c(1,1), srt=0, mai=c(0.6, 0.6, 0.4, 0.2), mgp=c(2,1,0))
trpSample=cbind(corareaSample[1:nIter],corareaSample[(nIter+1):(2*nIter)],corareaSample[(2*nIter+1):(3*nIter)])
for (i in 2:nIter) {
 trpSample[i,1]=((i-1)*trpSample[i-1]+corareaSample[i])/i
 trpSample[i,2]=((i-1)*trpSample[nIter+i-1]+corareaSample[nIter+i])/i
 trpSample[i,3]=((i-1)*trpSample[2*nIter+i-1]+corareaSample[2*nIter+i])/i
}
mn=mean(corareaSample); sd=sqrt(var(corareaSample))
plot(trpSample[,1],type='l',ylim=c(mn-sd,mn+sd),xlim=c(0,nIter),
 main='Trace plot of posterior mean for 3 chains',
 xlab='Draw',
 ylab='Mean')
lines(trpSample[,2],type='l',col='red')
lines(trpSample[,3],type='l',col='green')
dev.copy2eps(file=paste(outdir,"TracePlotIW4sim.eps",sep=""), width=6, height=6, horizontal=FALSE) #to file

#Autocorrelation plots
par(mfrow=c(1,1), srt=0, mai=c(0.6, 0.6, 0.4, 0.2), mgp=c(2,1,0))
autocorr.plot(corareaSample[1:nIter],main="MCMC Chain for p",auto.layout=FALSE)
dev.copy2eps(file=paste(outdir,"AutoCorrPlotIW4sim.eps",sep=""), width=6, height=6, horizontal=FALSE) #to file


## Generate statistical inferences and outputs ##

#Plots of drawn values
par(mfrow=c(3,2), srt=0, mai=c(0.6, 0.6, 0.4, 0.2), mgp=c(2,1,0))
plot(density(mcmcChain[,"be[1]"]),xlab="",ylab="Density",
 main="Intercept")
plot(density(mcmcChain[,"be[2]"]),xlab="",ylab="Density",
 main="Wave 2 effect")
plot(density(mcmcChain[,"be[3]"]),xlab="",ylab="Density",
 main="Wave 3 effect")
plot(density(mcmcChain[,"be[4]"]),xlab="",ylab="Density",
 main="Wave 4 effect")
plot(density(mcmcChain[,"corrand"]),xlab="",ylab="Density",
 main="Correlation for individual random effects")
plot(density(mcmcChain[,"corarea"]),xlab="",ylab="Density",
 main="Correlation for area random effects")
dev.copy2eps(file=paste(outdir,"FixedIW4sim.eps",sep=""), width=6, height=6, horizontal=FALSE) #to file

#Summary statistics of draws (the types of things you might put in a paper)
mn=apply(mcmcChain,2,mean)
stdev=sqrt(apply(mcmcChain,2,var))
lowl=apply(mcmcChain,2,quantile,0.025)
```

```
uppl=apply(mcmcChain,2,quantile,0.975)
sumdata=cbind(mn,stdev,lowl,uppl)
write.table(t(sumdata),file=paste(outdir,"BayesSummaryStatisticsIW4sim.txt",sep=""))
```

4.2 *JAGS code*

```
model {
   BMI[1] ~ dnorm(la[1],isigma.e[1]);
   la[1] <- pred[1];
   pred[1] <- be[1] + waveterms[1] + be[9]*male[1] + be[10]*cyob[1] + be[11]*smokes[1] + be[12]*married[1] + be[13]*educ1[1]
       + be[14]*educ2[1] + be[15]*employ[1] + be[16]*ctractpov[1] + be[17]*cetractpov[1]*employ[1]
       + (be[18]*female[1] + be[19]*male[1])*cDistHome[1] + (be[20]*female[1] + be[21]*male[1])*cDistWork[1]
       + (be[22]*female[1] + be[23]*male[1])*cDriveDist[1]
       + mu[ID[1], 1] + mu[ID[1], 2]*wave[1] + th[TractHome[1], 1] + th[TractWork[1], 2]*employ[1];
  isigma.e[1] <- isigma;

   waveterms[1] <- be[2]*wave2[1] + be[3]*wave3[1] + be[4]*wave4[1] + be[5]*wave5[1]
                   + be[6]*wave6[1] + be[7]*wave7[1] + be[8]*wave8[1];
   wave2[1] <- equals(wave[1], 2);
   wave3[1] <- equals(wave[1], 3);
   wave4[1] <- equals(wave[1], 4);
   wave5[1] <- equals(wave[1], 5);
   wave6[1] <- equals(wave[1], 6);
   wave7[1] <- equals(wave[1], 7);
   wave8[1] <- equals(wave[1], 8);
   educ1[1] <- equals(educ[1], 1);
   educ2[1] <- equals(educ[1], 2);
   female[1] <- 1 - male[1];
   employ[1] <- 1 - unemploy[1];
   cyob[1] <- yob[1] - mean(yob[]);
   ctractpov[1] <- tractpov[1] - mean(tractpov[]);
   cetractpov[1] <- (etractpov[1] - mean(etractpov[]))*employ[1];
   cDistHome[1] <- DistHome[1] - mean(DistHome[]);
   cDistWork[1] <- (DistWork[1] - mean(DistWork[]))*employ[1]; #Only non-zero for employed individuals
   cDriveDist[1] <- (DriveDist[1] - mean(DriveDist[]))*employ[1]; #Only non-zero for employed individuals
   Omni[1] ~ dunif(0,1);

   for (i in 2:length(ID)) {
      BMI[i] ~ dnorm(la[i],isigma.e[i]);
      la[i] <- pred[i] + rho*(BMI[i-1] - pred[i-1])*nfst[i];
      pred[i] <- be[1] + waveterms[i] + be[9]*male[i] + be[10]*cyob[i] + be[11]*smokes[i] + be[12]*married[i] + be[13]*educ1
          + be[14]*educ2[i] + be[15]*employ[i] + be[16]*ctractpov[i] + be[17]*cetractpov[i]*employ[i]
          + (be[18]*female[i] + be[19]*male[i])*cDistHome[i] + (be[20]*female[i] + be[21]*male[i])*cDistWork[i]
          + (be[22]*female[i] + be[23]*male[i])*cDriveDist[i]
          + mu[ID[i], 1] + mu[ID[i], 2]*wave[i] + th[TractHome[i], 1] + th[TractWork[i], 2]*employ[i];
      isigma.e[i] <- (1+ pow(rho, 2)*nfst[i])*isigma;
      nfst[i] <- step(Tobs[i]-1);

      waveterms[i] <- be[2]*wave2[i] + be[3]*wave3[i] + be[4]*wave4[i] + be[5]*wave5[i]
                      + be[6]*wave6[i] + be[7]*wave7[i] + be[8]*wave8[i];
      wave2[i] <- equals(wave[i], 2);
      wave3[i] <- equals(wave[i], 3);
      wave4[i] <- equals(wave[i], 4);
      wave5[i] <- equals(wave[i], 5);
      wave6[i] <- equals(wave[i], 6);
      wave7[i] <- equals(wave[i], 7);
      wave8[i] <- equals(wave[i], 8);
      educ1[i] <- equals(educ[i], 1);
```

```
   educ2[i] <- equals(educ[i], 2);
   female[i] <- 1 - male[i];
   employ[i] <- 1 - unemploy[i];
   cyob[i] <- yob[i] - mean(yob[]);
   ctractpov[i] <- tractpov[i] - mean(tractpov[]);
   cetractpov[i] <- (etractpov[i] - mean(etractpov[]))*employ[i];
   cDistHome[i] <- DistHome[i] - mean(DistHome[]);
   cDistWork[i] <- (DistWork[i] - mean(DistWork[]))*employ[i]; #Only non-zero for employed individuals
   cDriveDist[i] <- (DriveDist[i] - mean(DriveDist[]))*employ[i]; #Only non-zero for employed individuals
   Omni[i] ~ dunif(0,1);
}

#Distribution for random effects
for (j in 1:max(ID)) {
   mu[j,1:2] ~ dmnorm(mn1[], itaurand[,]);
}

#Distribution for area effects
for (k in 1:max(TractHome)) {
   th[k,1:2] ~ dmnorm(mn2[],itauarea[,]);
}

#Prior for fixed effects
for (k in 1:23) {
   be[k] ~ dnorm(0,1.0E-6);
}

#Hyper-priors
isigma ~ dgamma(1.0E-3,1.0E-3);
itaurand[1:2,1:2] ~ dwish(Omrand[,],df);
itauarea[1:2,1:2] ~ dwish(Omarea[,],df); #Make 2nd param = df
df <- 4 #2, 4 or 13 for second parameter
u ~ dbeta(1,1); #dbeta(1,1) = unif
rho <- 2*u - 1;

mn1[1] <- 0; mn1[2] <- 0;
mn2[1] <- 0; mn2[2] <- 0;
Omrand[1,1] <- 1; Omrand[1,2] <- 0; Omrand[2,1] <- 0; Omrand[2,2] <- 1;
Omarea[1,1] <- OmareaMn; Omarea[1,2] <- 0; Omarea[2,1] <- 0; Omarea[2,2] <- OmareaMn;
OmareaMn <- 1*(df - 3); #To ensure prior mean is 0.1, supply inverse of scaled mean to Wishart

#Inverse of variances and covariance matrices
sigma <- 1.0/isigma;
taurand[1:2,1:2] <- inverse(itaurand[,]);
tauarea[1:2,1:2] <- inverse(itauarea[,]);
corrand <- taurand[1,2]/sqrt(taurand[1,1]*taurand[2,2]);
corarea <- tauarea[1,2]/sqrt(tauarea[1,1]*tauarea[2,2]);

#Derived quantities of interest
iccind <- taurand[1,1]/(taurand[1,1]+sigma);
bothfdist <- (1 - step(be[18]))*(1 - step(be[20]));
onefdist <- 1 - step(be[18])*step(be[20]);
}
```

[*Received August 1, 2010; revised October 1, 2010; accepted for publication November 1, 2010*]