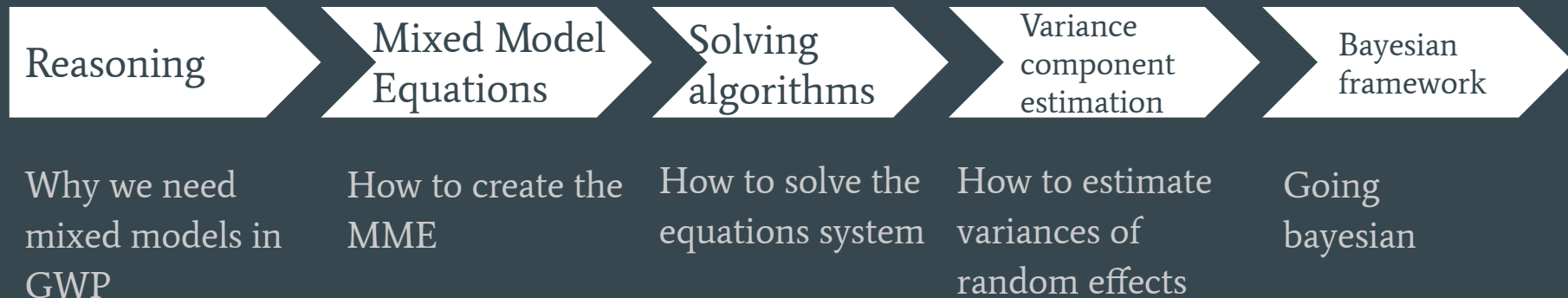


Linear Mixed Models

...

Evangelina López de Maturana & Oscar González-Recio

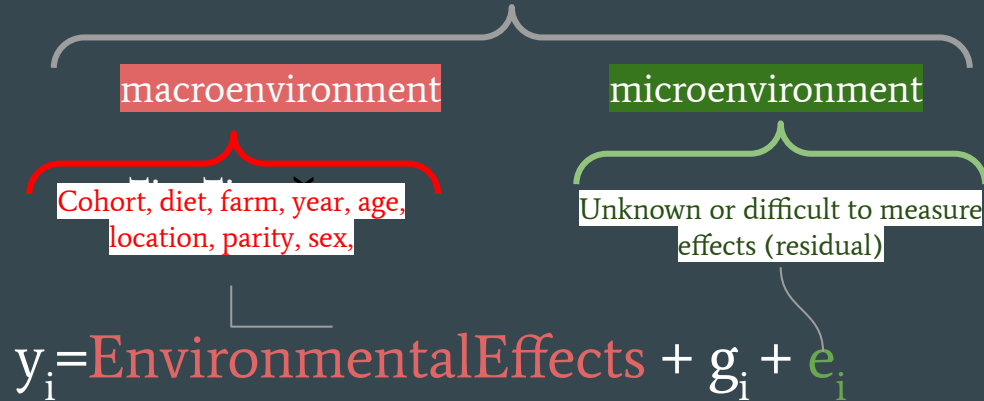
What you need to know



Mathematical representation of biological processes

$$P=G+E$$

$$y_i = g_i + E_i$$



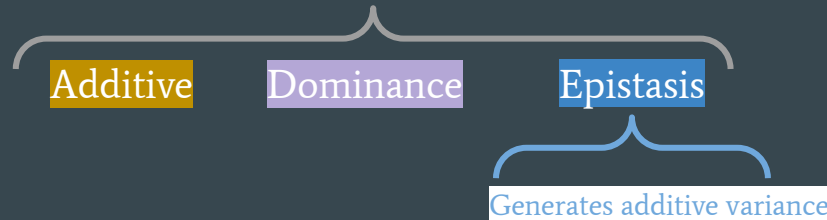
$$y_i = X_i b_i + Z_i g_i + e_i$$



Mathematical representation of biological processes

$$P=G+E$$

$$y_i = g_i + E_i$$



Hill et al (2008) <https://doi.org/10.1371/journal.pgen.1000008>



$$y_i = X_i b_i + Z_{ui} u_i + Z_{di} d_i + Z_{ei} u_i \# u_i + Z_{ei} u_i \# d_i + \dots + e_i$$

$$y_i = X_i b_i + Z_{ui} u_i + e'_i$$

Mathematical representation of biological processes

$$P=G+E$$

$$y_i = g_i + E_i$$

Additive

Dominance

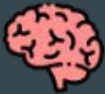
Epistasis

Generates additive variance

Hill et al (2008) <https://doi.org/10.1371/journal.pgen.1000008>

$$y_i = X_i b_i + Z_{ui} u_i + Z_{di} d_i + Z_{ei} u_i \# u_i + Z_{ei} u_i \# d_i + \dots + e_i$$

$$y_i = X_i b_i + Z_{ui} u_i + e'_i$$



Genome-wide prediction



Linear mixed models

$$y_i = X_i b_i + Z_{ui} u_i + e'_i$$

$$\mathbf{e} \sim N(0, \mathbf{I} \otimes \sigma_e^2)$$

$$\mathbf{u} \sim N(0, \mathbf{A} \otimes \sigma_a^2)$$

“Fixed” terms

“Random” terms (assume some known distribution to the effect)



Genome-wide prediction



Linear mixed models: Mixed Model Equation (EMM)

$$y_i = X_i b_i + Z_{ui} u_i + e'_i$$



$$\begin{pmatrix} Y_{111} \\ Y_{112} \\ Y_{113} \\ Y_{114} \\ Y_{115} \\ Y_{121} \\ Y_{122} \\ Y_{123} \\ Y_{124} \\ Y_{125} \\ Y_{211} \\ Y_{212} \\ Y_{213} \\ Y_{214} \\ Y_{215} \\ Y_{221} \\ Y_{222} \\ Y_{223} \\ Y_{224} \\ Y_{225} \\ Y_{311} \\ Y_{312} \\ Y_{313} \\ Y_{314} \\ Y_{315} \\ Y_{321} \\ Y_{322} \\ Y_{323} \\ Y_{324} \\ Y_{325} \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} \mu \\ \tau_1 \\ \tau_2 \end{pmatrix} + \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} h_1 \\ h_2 \\ h_3 \end{pmatrix} + \begin{pmatrix} e_{111} \\ e_{112} \\ e_{113} \\ e_{114} \\ e_{115} \\ e_{121} \\ e_{122} \\ e_{123} \\ e_{124} \\ e_{125} \\ e_{211} \\ e_{212} \\ e_{213} \\ e_{214} \\ e_{215} \\ e_{221} \\ e_{222} \\ e_{223} \\ e_{224} \\ e_{225} \\ e_{311} \\ e_{312} \\ e_{313} \\ e_{314} \\ e_{315} \\ e_{321} \\ e_{322} \\ e_{323} \\ e_{324} \\ e_{325} \end{pmatrix}$$

$Y = X \beta + Z u + e$

Linear mixed models: Mixed Model Equation (MME)

$$y_i = \mathbf{X}_i \mathbf{b}_i + \mathbf{Z}_{ui} \mathbf{u}_i + \mathbf{e}'_i$$

When \mathbf{X} and \mathbf{Z} are too large, inversion of matrices is very computationally demanding, there are linear combinations, and solving the system becomes cumbersome.

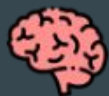
Henderson proposed MME

$$\begin{bmatrix} \mathbf{X}'\mathbf{X} & \mathbf{X}'\mathbf{Z} \\ \mathbf{Z}'\mathbf{X} & \mathbf{Z}'\mathbf{Z} + \alpha \mathbf{A}^{-1} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{\beta}} \\ \hat{\mathbf{u}} \end{bmatrix} = \begin{bmatrix} \mathbf{X}'\mathbf{Y} \\ \mathbf{Z}'\mathbf{Y} \end{bmatrix}$$

Linear mixed models: Mixed Model Equation (EMM)

Increased complexity

$$\mathbf{y}_i = \mathbf{X}_i \mathbf{b}_i + \mathbf{U}_i \mathbf{u}_i + \mathbf{W}_i \mathbf{p}_i + \mathbf{Z}_{ui} \mathbf{u}_i + \mathbf{e}'_i$$



$$\begin{pmatrix} \mathbf{X}'\mathbf{R}^{-1}\mathbf{X} & \mathbf{X}'\mathbf{R}^{-1}\mathbf{U} & \mathbf{X}'\mathbf{R}^{-1}\mathbf{W} & \mathbf{X}'\mathbf{R}^{-1}\mathbf{Z} \\ \mathbf{U}'\mathbf{R}^{-1}\mathbf{X} & \mathbf{U}'\mathbf{R}^{-1}\mathbf{U} + \Theta_1 & \mathbf{U}'\mathbf{R}^{-1}\mathbf{W} & \mathbf{U}'\mathbf{R}^{-1}\mathbf{Z} \\ \mathbf{W}'\mathbf{R}^{-1}\mathbf{X} & \mathbf{W}'\mathbf{R}^{-1}\mathbf{U} & \mathbf{W}'\mathbf{R}^{-1}\mathbf{W} + \Theta_2 & \mathbf{W}'\mathbf{R}^{-1}\mathbf{Z} \\ \mathbf{Z}'\mathbf{R}^{-1}\mathbf{X} & \mathbf{Z}'\mathbf{R}^{-1}\mathbf{U} & \mathbf{Z}'\mathbf{R}^{-1}\mathbf{W} & \mathbf{Z}'\mathbf{R}^{-1}\mathbf{Z} + \Theta_S \end{pmatrix}^{-1} \mathbf{X} \begin{pmatrix} \mathbf{X}'\mathbf{R}^{-1}\mathbf{Y} \\ \mathbf{U}'\mathbf{R}^{-1}\mathbf{Y} \\ \mathbf{W}'\mathbf{R}^{-1}\mathbf{Y} \\ \mathbf{Z}'\mathbf{R}^{-1}\mathbf{Y} \end{pmatrix}$$



Genome-wide prediction

100110
00101
10101

Linear mixed models: Mixed Model Equation (EMM)

Marker regression

$$y_i = \mathbf{X}_i \mathbf{b}_i + \mathbf{Z}_{ui} \mathbf{u}_i + e'_i$$

$$\mathbf{y} = \mu \mathbf{1} + \mathbf{X} \boldsymbol{\beta}_f + \textit{snp}_1 \beta_1 + \textit{snp}_2 \beta_2 + \dots + \textit{snp}_p \beta_p + \mathbf{e}$$

$$\mathbf{e} \sim N(0, \sigma_e^2)$$

$$\mathbf{b}_i \sim N(0, \sigma_i^2)$$

$$\sigma_i^2 \sim \chi_{(v,S)}^{-2}$$



Genome-wide prediction



Variance component estimation

Methods

Estimate residual, genetic, permanent, marker, ... variances

- ANOVA
- Maximum Likelihood (ML)
- Restricted Maximum Likelihood (REML)
- Minimum Norm Quadratic Unbiased Estimation (MINQUE I, II, III)
- Minimum Variance Quadratic Unbiased Estimation (MIVQUE)



Journal of the American Statistical Association

Publication details, including instructions for authors and subscription information:

<http://www.tandfonline.com/loi/uasa20>

A Comparison of Variance Component Estimates for Arbitrary Underlying Distributions

Peter H. Westfall^a

^a Department of Information Systems and Quantitative Sciences, College of Business Administration, Texas Tech University, Lubbock, TX, 79409, USA
Published online: 12 Mar 2012.

Solve MME

Algorithms

- Gauss-seidel
- Choleski decomposition
- Preconditioned conjugate gradients (PCG)
- Gauss-seidel with residual updates

J. Dairy Sci. 91:360–366
doi:10.3168/jds.2007-0403

© American Dairy Science Association, 2008.

Technical Note: Computing Strategies in Genome-Wide Selection

A. Legarra^{*1} and I. Misztal[†]

^{*}Institut National de la Recherche Agronomique, UR631 Station d'Amélioration Génétique des Animaux, BP 52627,
32326 Castanet-Tolosan, France

[†]Department of Animal and Dairy Science, University of Georgia, Athens 30602-2771

Solve MME

Algorithms

2,000 records
11,000 SNPs

Table 1. Computing times for different methods of solving the mixed-model equations in a case of genome-wide genetic evaluation¹

Item	Cholesky decomposition ²	LHS-GS	Matrix-free GS	Matrix-free GSRU	Matrix-free PCG
Convergence = 10^{-10}					
Time to set up	17 min	17 min	16 s	16 s	20 s
Time for solving	119 min	71 min	97 h	46 s	10 s
Number of iterations	1	164	161	164	20
Convergence = 10^{-14}					
Time to set up	17 min	17 min	³	16 s	20 s
Time for solving	119 min	170 min	³	74 s	12 s
Number of iterations	1	272	³	272	23

¹LHS = left-hand side; GS = Gauss-Seidel; GSRU = Gauss-Seidel with residual update; PCG = preconditioned conjugated gradients.

²This is an exact method and the convergence measure is meaningless.

³Not tried.

Solve MME

Gauss Seidel with Residual Update

y corrected for all effects except the j^{th} effect, is equal to the current vector of residuals plus the current estimate of the j^{th} effect.

Then, we can compute this j^{th} effect.

We update the residuals with the new j^{th} effect solution.

$\mathbf{x}'_j \mathbf{x}_j$ are constant, so they can be precomputed, updating vector products and residuals at each iteration, speeding up the algorithm

$$\mathbf{y} - \mathbf{X}_{1:j-1} \hat{\mathbf{a}}_{1:j-1}^{l+1} - \mathbf{X}_{j+1:n} \hat{\mathbf{a}}_{j+1:n}^l = \mathbf{e}^{l+1,j} + \mathbf{x}_j \hat{a}_j^l.$$

Then

$$\hat{a}_j^{l+1} = \frac{\mathbf{x}'_j \mathbf{e}^{l+1,j} + \mathbf{x}'_j \mathbf{x}_j \hat{a}_j^l}{\mathbf{x}'_j \mathbf{x}_j + \lambda}.$$

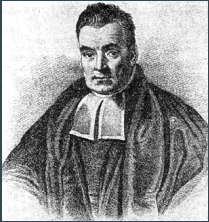
$$\mathbf{e}^{l+1,j+1} = \mathbf{e}^{l+1,j} - \mathbf{x}_j (\hat{a}_j^{l+1} - \hat{a}_j^l).$$



Bayesian framework

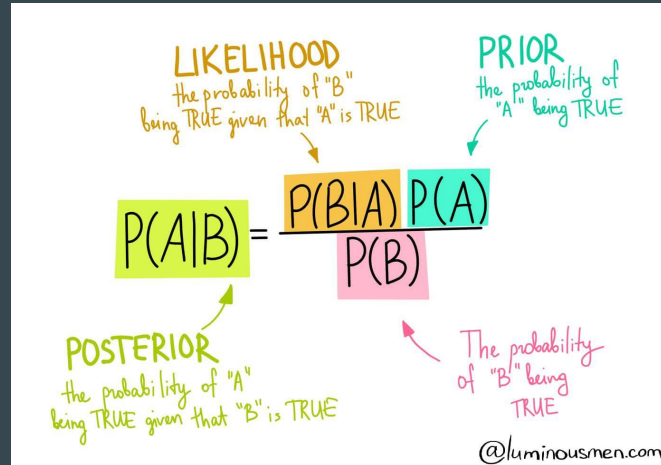
Assume a model for the data

Bayes theorem



$$\mathbf{y} = \mu \mathbf{1} + \mathbf{X}\boldsymbol{\beta}_f + \text{sn}p_1\boldsymbol{\beta}_1 + \text{sn}p_2\boldsymbol{\beta}_2 + \dots + \text{sn}p_p\boldsymbol{\beta}_p + \mathbf{e}$$

$$p(\mathbf{y}|\boldsymbol{\theta}) = N(\mathbf{X}\boldsymbol{\beta} + \dots + \mathbf{Z}\mathbf{u}, \sigma_e^2)$$



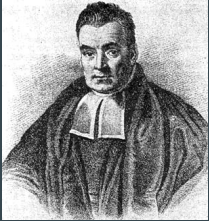
Bayesian framework

Assume a model for the data

$$p(\mathbf{y}|\boldsymbol{\theta}) = N(\mathbf{X}\boldsymbol{\beta} + \cdots + \mathbf{Z}\mathbf{u}, \sigma_e^2)$$

Bayes theorem

$$p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta}) = p(\boldsymbol{\theta}|\mathbf{y})p(\mathbf{y})$$

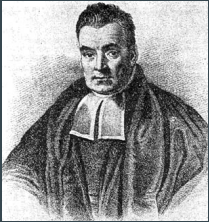


Bayesian framework

Assume a model for the data

$$p(\mathbf{y}|\boldsymbol{\theta}) = N(\mathbf{X}\boldsymbol{\beta} + \dots + \mathbf{Z}\mathbf{u}, \sigma_e^2)$$

Bayes theorem



...proportionality

$$p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta}) = p(\boldsymbol{\theta}|\mathbf{y})p(\mathbf{y})$$

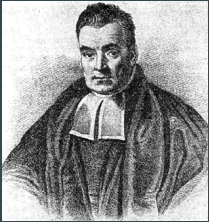
$$p(\boldsymbol{\theta}|\mathbf{y}) = \frac{p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathbf{y})} \propto p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})$$

Bayesian framework

Assume a model for the data

$$p(\mathbf{y}|\boldsymbol{\theta}) = N(\mathbf{X}\boldsymbol{\beta} + \dots + \mathbf{Z}\mathbf{u}, \sigma_e^2)$$

Bayes theorem



$$p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta}) = p(\boldsymbol{\theta}|\mathbf{y})p(\mathbf{y})$$

...proportionality

$$p(\boldsymbol{\theta}|\mathbf{y}) = \frac{p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathbf{y})} \propto p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})$$

Choose priors

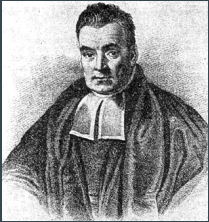
$$p(\boldsymbol{\theta}|\mathbf{y}) \propto p(\mathbf{y}|\boldsymbol{\beta}, \mathbf{u}, \sigma_e^2)p(\boldsymbol{\beta}|\sigma_b^2)p(\mathbf{u}|\sigma_u^2)p(\sigma_b^2)p(\sigma_u^2)p(\sigma_e^2)$$

Bayesian framework

Assume a model for the data

$$p(\mathbf{y}|\boldsymbol{\theta}) = N(\mathbf{X}\boldsymbol{\beta} + \dots + \mathbf{Z}\mathbf{u}, \sigma_e^2)$$

Bayes theorem



$$p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta}) = p(\boldsymbol{\theta}|\mathbf{y})p(\mathbf{y})$$

...proportionality

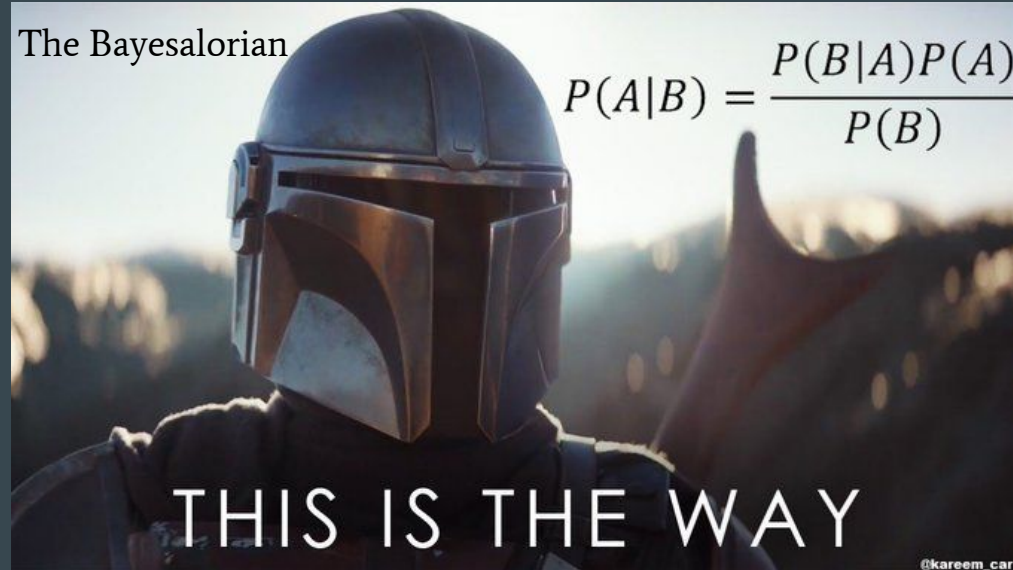
$$p(\boldsymbol{\theta}|\mathbf{y}) = \frac{p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathbf{y})} \propto p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})$$

Choose priors

$$p(\boldsymbol{\theta}|\mathbf{y}) \propto p(\mathbf{y}|\boldsymbol{\beta}, \mathbf{u}, \sigma_e^2)p(\boldsymbol{\beta}|\sigma_b^2)p(\mathbf{u}|\sigma_u^2)p(\sigma_b^2)p(\sigma_u^2)p(\sigma_e^2)$$

Make inferences using MCMC algorithms (Gibbs sampling, acceptance rejection, Metropolis-Hasting)

Bayesian framework



EXAMPLE

Create MME and solve the system using residual updates

ID	trait	Diet	Cohort
1	20	1	1
2	25	1	1
3	30	2	1
4	35	2	2
5	20	3	2
6	30	3	2

HOMEWORK

Solve the mixed linear model

$\text{trait} = \mu + \text{age} + \text{Diet} + \text{Cohort} + e$

Using Gauss-Seidel with residual updates, with residual variance = 40 and cohort variance = 30.

ID	trait	Age	Diet	Cohort
1	93	25	1	1
2	90	30	1	2
3	115	35	2	1
4	110	20	2	2
5	87	22	3	3
6	70	29	1	3
7	100	31	3	3

