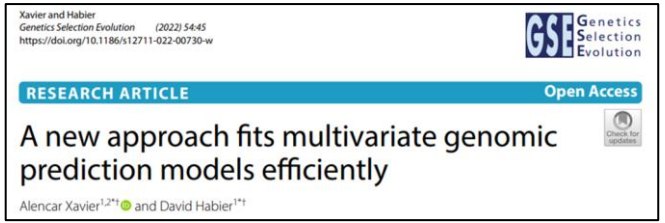
An Introduction to Scalable Multivariate & Megavariate Models

Alencar Xavier
Breeding Analyst at Corteva
Adjunct professor at Purdue
https://alenxav.github.io/

REFERENCES: https://doi.org/10.1186/s12711-022-00730-w and https://doi.org/10.1093/genetics/iyae179

Outline

1. **Introduction**
   - Rationale and statistical model
2. **Coefficients**
   - Univariate
   - Multivariate
3. **Variances**
   - Univariate
   - Multivariate
4. **Simulations**
   - Elapsed time
   - Benchmarks
5. **Megavariate**
   - Framework
   - Benchmarks
6. **Conclusion**

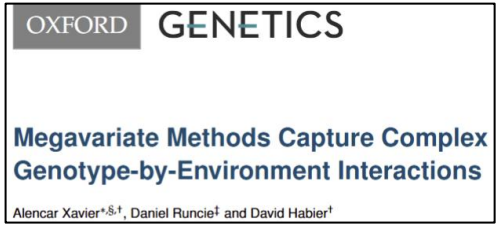

Xavier and Habier
Genetics Selection Evolution (2022) 54:45
https://doi.org/10.1186/s12711-022-00730-w

GSE Genetics Selection Evolution

RESEARCH ARTICLE Open Access

A new approach fits multivariate genomic prediction models efficiently

Alencar Xavier[1,2*†] and David Habier[1*†]


OXFORD GENETICS

Megavariate Methods Capture Complex Genotype-by-Environment Interactions

Alencar Xavier[*,§,†], Daniel Runcie[‡] and David Habier[†]

1. **Introduction**
   - Rationale and statistical model
2. **Coefficients**
   - Univariate
   - Multivariate
3. **Variances**
   - Univariate
   - Multivariate
4. **Simulations**
   - Elapsed time
   - Benchmarks
5. **Megavariate**
   - Framework
   - Benchmarks
6. **Conclusion**

What is genomic prediction?

DATASET	GENOTYPES	PHENOTYPES
TRAINING POPULATION	YES	YES
PREDICTION TARGET	YES	NO

Purpose of GS:

- Improve selection accuracy
- Select material without phenotypes
- Selection of new parents
- Prediction of cross combinations
- Optimize resources
- Stability and genetic architecture

Rationale

- Single-trait models for genomic prediction in plant breeding are already well-stablished (*e.g.* GBLUP and BayesB)

- Phenotypes come from multiple locations, years, and quantitative traits; and most traits have genetically correlated breeding values

Rationale

- **Complex GxE patterns / multi-trait** = higher accuracy

- **Assess new phenomic traits** (*e.g.* canopy coverage in soy)

- **Computationally PROHIBITIVE***

* Zhou, X., & Stephens, M. (2014). Efficient multivariate linear mixed model algorithms for genome-wide association studies. *Nature methods*, *11*(4), 407-409.

Practical example

Fit a model using phenotypes (Y) and genotypes (X)

```
> require(bWGR)
> fit = mrr(Y,X)
> round(fit$h2,2)
[1] 0.38 0.48 0.71 0.63 0.60
> round(fit$GC,2)
     [,1] [,2] [,3] [,4] [,5]
[1,] 1.00 0.76 0.70 0.64 0.62
[2,] 0.76 1.00 0.56 0.65 0.39
[3,] 0.70 0.56 1.00 0.71 0.23
[4,] 0.64 0.65 0.71 1.00 0.24
[5,] 0.62 0.39 0.23 0.24 1.00
```

Genomic heritability

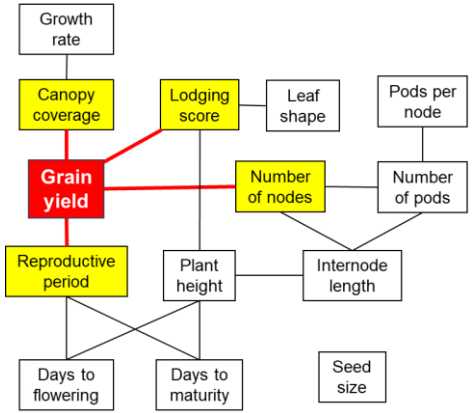
Genetic correlations

+ Genomic breeding values to make selections
+ Marker effects to predict new individuals
+ Variance components to create selection indices

Multivariate models enable analysis of

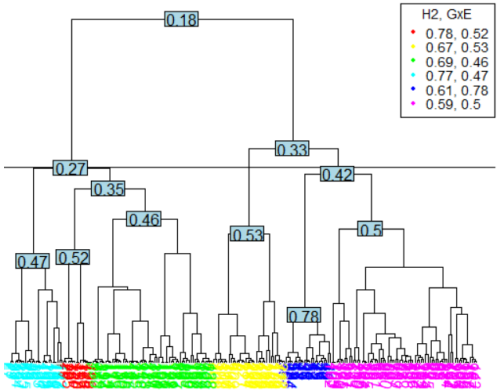
Multiple traits

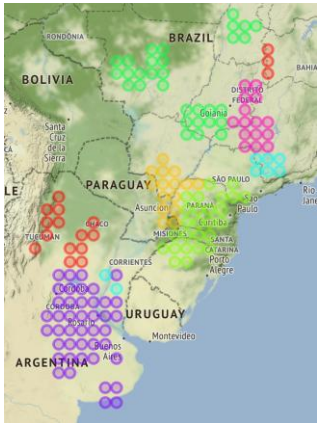
Graph derived from the genetic correlation among soybean traits
https://rd.springer.com/article/10.1007/s10681-017-1975-4



Multiple environments

Example of environmental clustering





Why would multivariate be any better?

Simple (bivariate) model:

INFORMATION GAIN

$$\mathbf{y} = \mathbf{g} + \mathbf{e}$$

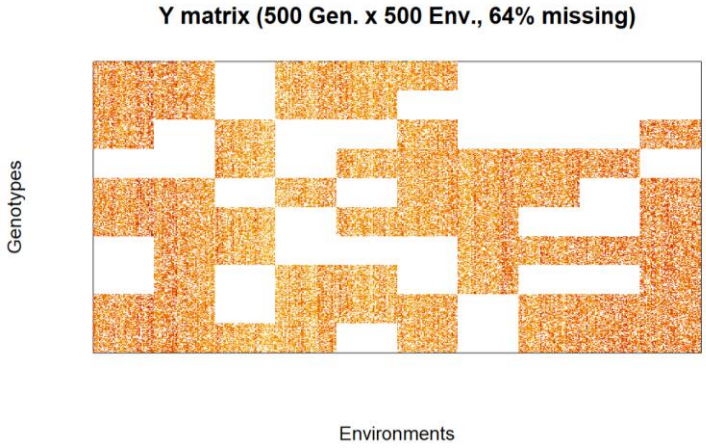
$$Var\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} \sigma^2_{a_1} & \sigma_{a_{12}} \\ \sigma_{a_{12}} & \sigma^2_{a_2} \end{bmatrix} + \begin{bmatrix} \sigma^2_{e_1} & \sigma_{e_{12}} \\ \sigma_{e_{12}} & \sigma^2_{e_2} \end{bmatrix}$$

What is a megavariate model?

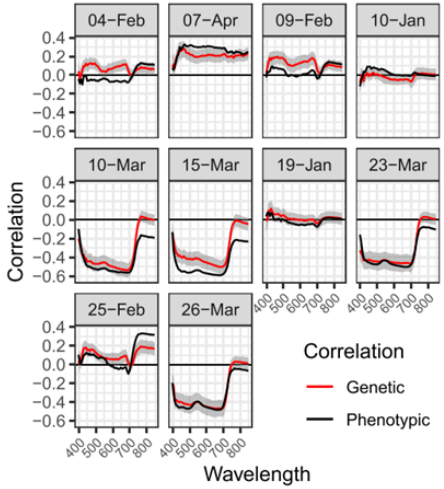
Model:

$$\mathbf{Y} = \mathbf{G} + \mathbf{E}$$
$$= \mathbf{XB} + \mathbf{E}$$
$$\mathbf{B} \sim \mathbf{N}(\mathbf{0}, \mathbf{\Sigma_G})$$

- Large number of environments
- High-throughput phenotyping



Simulated example of sparse multi-environmental trials



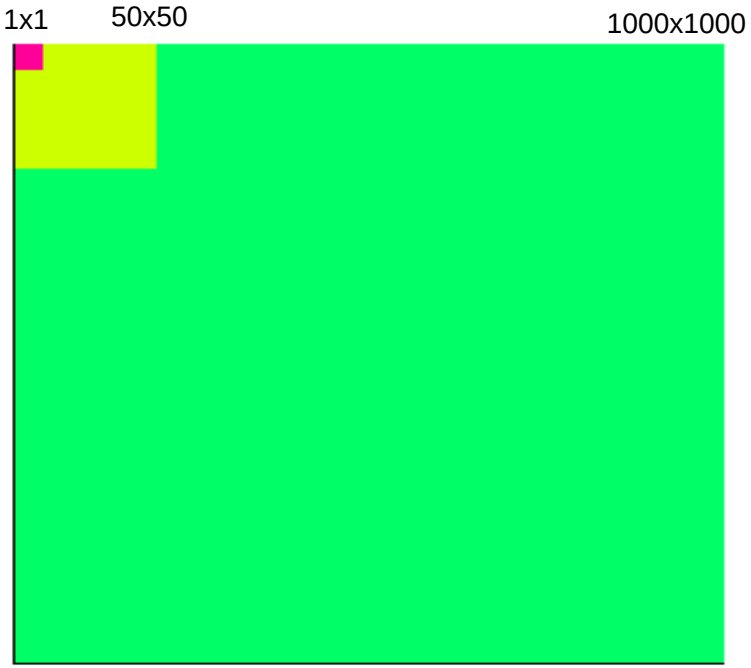
Runcie et al. (2021) Mega-scale linear mixed models for genomic predictions with thousands of traits. https://doi.org/10.1186/s13059-021-02416-w

Scale of Σ_β

Megavariate
50+ traits

Multivariate
2-50 traits

Univariate



Multivariate computational complexity is exponential k^7 (Zhou and Stephens 2014)

Multivariate Statistical Model

$$y = \mu + \mathbf{Z}\beta + e \qquad (1)$$

- Where $y = \{y_1, y_2, \dots, y_K\}$, $\mu = \{\mu_1, \mu_2, \dots, \mu_K\}$, $\beta = \{\beta_1, \beta_2, \dots, \beta_K\}$,

$$e = \{e_1, e_2, \dots, e_K\},\ Z = \mathrm{BlockDiag}\{\mathbf{Z}_1, \mathbf{Z}_2, \dots, \mathbf{Z}_K\}$$

- Variances:

$$\Sigma_\beta = \begin{bmatrix} \sigma^2_{\beta(1)} & \cdots & \sigma_{\beta(1,K)} \\ \vdots & \ddots & \vdots \\ \sigma_{\beta(K,1)} & \cdots & \sigma^2_{\beta(K)} \end{bmatrix} \text{ and } \Sigma_e = \begin{bmatrix} \sigma^2_{e(1)} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma^2_{e(K)} \end{bmatrix}$$

Corresponding mixed model equation

Under the traditional framework, the mixed-model equations required to solve the multivariate ridge regression (eq. 1) can be written as follows:

$$\begin{bmatrix} \mathbf{1}_1'\mathbf{1}_1\sigma_{e_1}^{-2} & \dots & \mathbf{0} & \mathbf{1}_1'\mathbf{Z}_1\sigma_{e_1}^{-2} & \dots & \mathbf{0} \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{0} & \dots & \mathbf{1}_K'\mathbf{1}_K\sigma_{e_K}^{-2} & \mathbf{0} & \dots & \mathbf{1}_K'\mathbf{Z}_K\sigma_{e_K}^{-2} \\ \mathbf{Z}_1'\mathbf{1}_1'\sigma_{e_1}^{-2} & \dots & \mathbf{0} & \mathbf{Z}_1'\mathbf{Z}_1\sigma_{e_1}^{-2} + \mathbf{I}_m\sigma_\beta^{11} & \dots & \mathbf{I}_m\sigma_\beta^{1K} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \dots \\ \mathbf{0} & \dots & \mathbf{Z}_K'\mathbf{1}_K'\sigma_{e_K}^{-2} & \mathbf{I}_m\sigma_\beta^{K1} & \vdots & \mathbf{Z}_K'\mathbf{Z}_K\sigma_{e_K}^{-2} + \mathbf{I}_m\sigma_\beta^{KK} \end{bmatrix} \begin{bmatrix} \hat{\mu}_1 \\ \vdots \\ \hat{\mu}_k \\ \hat{\boldsymbol{\beta}}_1 \\ \vdots \\ \hat{\boldsymbol{\beta}}_K \end{bmatrix} = \begin{bmatrix} \sigma_{e1}^{-2}\mathbf{1}_1'\boldsymbol{y}_1 \\ \vdots \\ \sigma_{e_K}^{-2}\mathbf{1}_k'\boldsymbol{y}_K \\ \sigma_{e1}^{-2}\mathbf{Z}_1'\boldsymbol{y}_1 \\ \vdots \\ \sigma_{e_K}^{-2}\mathbf{Z}_k'\boldsymbol{y}_K, \end{bmatrix} \quad (2)$$

where σ_β^{ij} is the element at position ij of $\boldsymbol{\Sigma}_\beta^{-1}$. This setup involves storing K times the cross-product or marker scores ($\mathbf{Z}_k'\mathbf{Z}_k$), each with dimension $m \times m$.

Moreover, this **huge** matrix must be **inverted** for the estimation of covariance components: $\hat{\Sigma}_{\beta(i,j)} = m^{-1}[\hat{\boldsymbol{\beta}}_i'\hat{\boldsymbol{\beta}}_j + tr(\mathbf{C}^{ij})]$

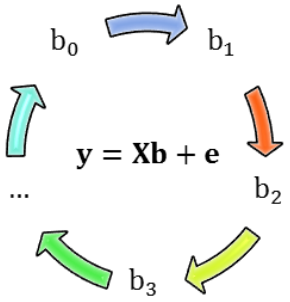
Computing very large multivariate models is **impossible**

unless…

1. **Introduction**
   - Rationale and statistical model
2. **Coefficients**
   - Univariate
   - Multivariate
3. **Variances**
   - Univariate
   - Multivariate
4. **Simulations**
   - Elapsed time
   - Benchmarks
5. **Megavariate**
   - Framework
   - Benchmarks
6. **Conclusion**

Coefficients for univariate model

1. Whole-genome regression (*e.g.* BayesA) rely on the *Gauss-Seidel* method [1]
2. GS has only two steps, whereas coordinate descent has three [2]
3. It avoids building the systems of equations altogether!!
4. Fits one marker effects, then uses residuals to fit the next

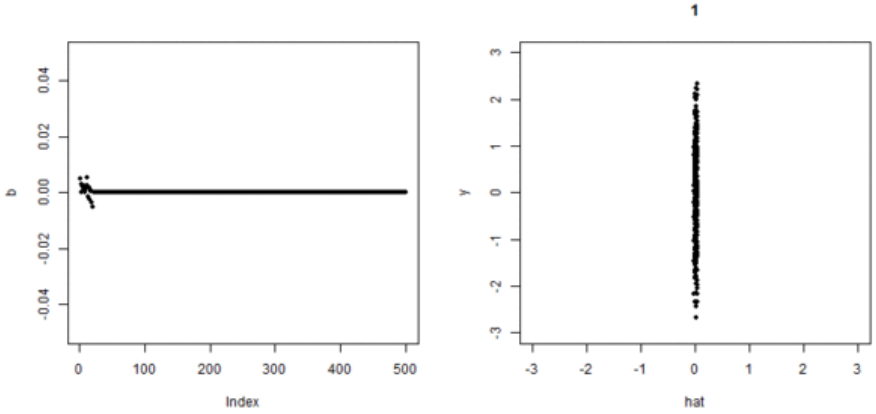


for j in 1:p {

$$\hat{b}_j^{t+1} = \frac{x_j'\hat{e}^t + x_j'x_j\hat{b}_j^t}{x_j'x_j + \lambda}$$

$$\hat{e}^{t+1} = \hat{e}^t - x_j\left(\hat{b}_j^{t+1} - \hat{b}_j^t\right)$$

}



1 Legarra, A., & Misztal, I. (2008). Computing strategies in genome-wide selection. *Journal of dairy science*, *91*(1), 360-366.
2 Xavier, A. (2021). Technical nuances of machine learning. *Crop Breeding and Applied Biotechnology*, 21.

Coefficients for <u>multivariate</u> model

For updating estimated marker effects we define, $\hat{\boldsymbol{\beta}}_j^{'(t)} = [\hat{\beta}_{j1}^{(t)}\ \hat{\beta}_{j1}^{(t)}\ \ldots\ \hat{\beta}_{jK}^{(t)}]$ to be the vector of estimated marker effects for marker j and all K environments, $\mathbf{Z}_j = \oplus_{k=1}^{K} \boldsymbol{z}_{jk}$ to be a matrix containing marker scores at marker j, and $\hat{\boldsymbol{\Sigma}}_e^{(t)} = Diag\{\hat{\sigma}_{e1}^{2(t)}, \hat{\sigma}_{e2}^{2(t)}, \ \ldots\ , \hat{\sigma}_{ek}^{2(t)}\}$ to be a diagonal matrix of estimated residual variances. Effects for marker j are initialized with zero and updated as

$$\hat{\boldsymbol{\beta}}_j^{(t+1)} = (\hat{\boldsymbol{\Sigma}}_e^{-1(t)}\mathbf{Z}_j'\mathbf{Z}_j + \hat{\boldsymbol{\Sigma}}_\beta^{-1(t)})^{-1}\mathbf{Z}_j'\hat{\boldsymbol{\Sigma}}_e^{-1(t)}(\mathbf{Z}_j\hat{\boldsymbol{\beta}}_j^{(t)} + \hat{\boldsymbol{e}}^{(t)}), \tag{5}$$

and before moving to the next marker, the residual vector is updated as

$$\hat{\boldsymbol{e}}^{(t+1)} = \hat{\boldsymbol{e}}^{(t)} - \mathbf{Z}_j'(\hat{\boldsymbol{\beta}}_j^{(t+1)} - \hat{\boldsymbol{\beta}}_j^{(t)}). \tag{6}$$

Note that the computation of Kronecker products are not necessary for the multivariate Gauss-Seidel formulation (eq. 5) as long as the residual covariance $\hat{\boldsymbol{\Sigma}}_e$ is a diagonal matrix.

NO KRONECKER PRODUCTS!!!!

For(j in 1:p) {

These genetic covariances are the whole key for the MRR model

1st solve for beta

$$\begin{bmatrix} \hat{\boldsymbol{\Sigma}}_{\beta}^{11} + \mathbf{z}'_{j(1)}\mathbf{z}_{j(1)}\sigma_{e(1)}^{-2} & \hat{\boldsymbol{\Sigma}}_{\beta}^{12} \\ \hat{\boldsymbol{\Sigma}}_{\beta}^{21} & \hat{\boldsymbol{\Sigma}}_{\beta}^{22} + \mathbf{z}'_{j(2)}\mathbf{z}_{j(2)}\sigma_{e(2)}^{-2} \end{bmatrix} \begin{bmatrix} \hat{\beta}_{j(1)}^{t+1} \\ \hat{\beta}_{j(2)}^{t+1} \end{bmatrix} = \begin{bmatrix} \sigma_{e(1)}^{-2}\left(\mathbf{z}'_{j(1)}\mathbf{z}_{j(1)}\hat{\beta}_{j(1)}^{t} + \mathbf{z}'_{j(1)}\hat{e}_{1}^{t}\right) \\ \sigma_{e(2)}^{-2}\left(\mathbf{z}'_{j(2)}\mathbf{z}_{j(2)}\hat{\beta}_{j(2)}^{t} + \mathbf{z}'_{j(2)}\hat{e}_{2}^{t}\right) \end{bmatrix}$$

2nd update residuals

$$\begin{bmatrix} \hat{e}_{j(1)}^{t+1} \\ \hat{e}_{j(2)}^{t+1} \end{bmatrix} = \begin{bmatrix} \hat{e}_{1}^{t} + \mathbf{z}'_{j(1)}\left(\hat{\beta}_{j(1)}^{t+1} - \hat{\beta}_{j(1)}^{t}\right) \\ \hat{e}_{2}^{t} + \mathbf{z}'_{j(2)}\left(\hat{\beta}_{j(2)}^{t+1} - \hat{\beta}_{j(2)}^{t}\right) \end{bmatrix}$$

}

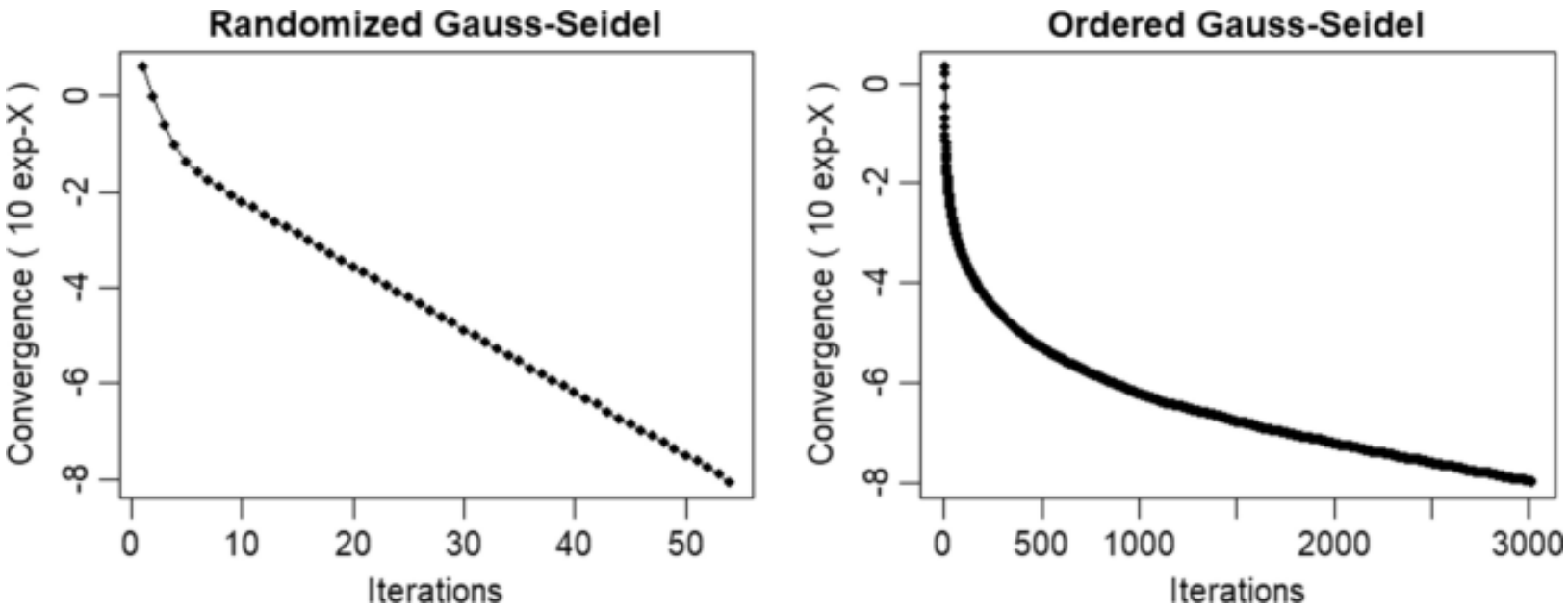
Color code

- **Computed only once, before the loop starts (ZpZ)**
- **Computed once every iteration**
- **Computed for each marker in every iteration**

What is in memory?

- Z (n x m)
- B (m x k)
- E (n x k)
- ZpZ (m x k)
- $\hat{\boldsymbol{\Sigma}}_{\beta}^{-1}$ (k x k)
- $\hat{\boldsymbol{\Sigma}}_{e}^{-1}$ (k)

Side note: Updating markers in random order can speed up convergence



Convergence of the Gauss–Seidel solver with (left) and without (right) randomizing the order in which marker effects were updated for one replicate of the simulation of scenario 2

1. **Introduction**
   - Rationale and statistical model
2. **Coefficients**
   - Univariate
   - Multivariate
3. **Variances**
   - Univariate
   - Multivariate
4. **Simulations**
   - Elapsed time
   - Benchmarks
5. **Megavariate**
   - Framework
   - Benchmarks
6. **Conclusion**

Variance components

Airy (1861): PRE-ANOVA → Fisher (1921): ANOVA

Henderson Methods' 1, 2 and 3 (Henderson 1953)

Restricted likelihood (Thompson 1962)

Iterative path

Exact Likelihood path

Iterative methods (Cunningham and Henderson 1968)

Hartley & Rao (1967): Maximum Likelihood

Henderson's Simple Method HSM (Henderson 1980)

Iterative HSM (Hudson and VanVleck 1982)

Non-orthogonal iterative method (Thompson 1969)

MINQUE (RAO 1970)

MIVQUE (RAO 1978)

≡

REML (Patterson and Thompson 1971)

Pseudo-Expectation "PE" (Schaeffer 1986)

Tilde-Hat "TH" (VanRaden and Jung 1987)

EM-REML (Henderson 1973)

DF-REML (Graser et al 1987)

AI-REML (Johnson and Thompson 1994)

Fast variations

Gold standard

Univariate case: Variance components

- REML

$$\frac{\partial LL}{\partial \hat{\sigma}^2_\beta} = 0 \rightarrow \hat{\sigma}^2_\beta = \frac{y'S'V^{-1}ZZ'V^{-1}Sy}{tr(V^{-1}ZZ')} = \frac{\hat{\beta}'\hat{\beta}}{tr(V^{-1}\tilde{Z}'\tilde{Z})}$$

All methods yield the same residual variance:

$$\hat{\sigma}^2_e = \frac{y'e}{n-1}$$

DA Harville 1977

"Let's get rid of this V^{-1}!"

- Schaffer's (Thompson's) Pseudo-Expectation

$$\hat{\sigma}^2_\beta = \frac{y'S'\cancel{V^{-1}}ZZ'V^{-1}Sy}{tr(\cancel{V^{-1}}SZZ'S)} = \frac{\tilde{y}'Z\hat{\beta}}{tr(\tilde{Z}'\tilde{Z})}$$

V is a pain to compute

$$\mathbf{V = ZZ'\sigma^2_\beta + I\sigma^2_\beta}$$

$S = I - (X'X)^{-1}X'$; $\quad P = V^{-1}S$

$P = V^{-1} - V^{-1}(X'V^{-1}X)^{-1}X'V^{-1}$

$PX = SX = 0$

$Sy = \text{Centralized } y = \tilde{y}$

$SZ = \text{Centralized } Z = \tilde{Z}$

"Let's replace this V^{-1} by something similar, but easier to compute!"

- VanRaden's Tilde-Hat

$$\hat{\sigma}^2_\beta = \frac{y'S'D^{-1}ZZ'V^{-1}Sy}{tr(D^{-1}SZZ')} = \frac{\overbrace{\tilde{y}D^{-1}Z\hat{\beta}}^{\tilde{\boldsymbol{\beta}}}}{tr(D^{-1}\tilde{Z}'\tilde{Z})} = \frac{\tilde{\beta}\hat{\beta}}{tr(D^{-1}\tilde{Z}'\tilde{Z})}$$

$$D = Diag(Z'Z\hat{\sigma}^{-2}_e + I\hat{\sigma}^{-2}_\beta)$$

Multivariate case: (co)variance components

$$\hat{\sigma}^2_{\beta(k)} = \frac{\tilde{\beta}_k \hat{\beta}_k}{tr(D_k^{-1} \tilde{Z}_k{}' \tilde{Z}_k)} \qquad \hat{\sigma}_{\beta(k,k')} = \frac{\tilde{\beta}_k \hat{\beta}_{k'} + \tilde{\beta}_{k'} \hat{\beta}_k}{tr(D_k^{-1} \tilde{Z}_k{}' \tilde{Z}_k) + tr(D_{k'}^{-1} \tilde{Z}_{k'}{}' \tilde{Z}_{k'})}$$

$$\hat{\sigma}^2_{e(k)} = \frac{y_k' \hat{e}_k}{n_k - 1}$$

Note: Schaffer's is obtained by assuming $\mathbf{D} = \mathbf{I}$

No V, No C, No LHS,
No determinants,
No dense inversions

Color code

- **Computed only once, before the loop starts (ZpZ)**
- **Computed once every iteration**
- **Computed once for PE, and every iteration for TH**

What is in memory?

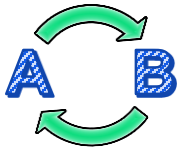
- Z (n x m)
- Bhat (m x k)
- Btilde (m x k)
- E (n x k)
- Y (n x k)
- Ytilde (n x k)
- ZpZ (m x k)
- ZpZtilde (m x k)
- $\hat{\Sigma}_\beta$ (k x k)
- $\hat{\Sigma}_e$ (k)
- N (k)

An intuitive derivation for Schaeffer's method?

The genetic covariance is simply estimated as the <u>cross-prediction between traits A and B</u> normalized by the scale of Zs

Centered phenotype of **A** | **A** predicted from **B** | Centered phenotype of **B** | **B** predicted from **A**

$$\hat{\sigma}_{\beta(A,B)} = \frac{(\mathbf{y}_A - \mu_A)'(\mathbf{Z}_A\boldsymbol{\beta}_B) + (\mathbf{y}_B - \mu_B)'(\mathbf{Z}_B\boldsymbol{\beta}_A)}{\mathrm{Tr}\left(\tilde{\mathbf{Z}}_A{}'\tilde{\mathbf{Z}}_A\right) + \mathrm{Tr}\left(\tilde{\mathbf{Z}}_B{}'\tilde{\mathbf{Z}}_B\right)}$$



1. **Introduction**
   - Rationale and statistical model
2. **Coefficients**
   - Univariate
   - Multivariate
3. **Variances**
   - Univariate
   - Multivariate
4. **Simulations**
   - Elapsed times
   - Benchmarks
5. **Megavariate**
   - Framework
   - Benchmarks
6. **Conclusion**

Metrics

1. Computation efficiency:

   Elapsed time to fit the model

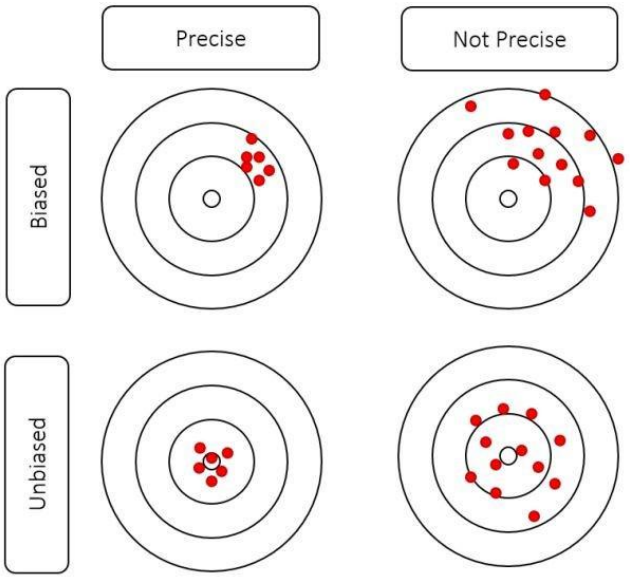
2. Breeding values:

   $$\text{Accuracy} = \text{cor}(\text{GEBV}, \text{TBV})$$

3. Heritability (h^2) and genetic correlations (ρ):

   $$\text{Bias} = \text{E}(\hat{\theta} - \theta)$$

   $$\text{Precision} = \text{SD}(\hat{\theta} - \theta)$$



Picture source

Datasets

	Small Balanced	Large Unbalanced
	Scenario 1	**Scenario 2**
Number of environments (traits)	10	10
Number of environments per line	10	1
Number of lines per environment	599	514
% of lines per environment	100%	10%
Number of phenotypic records	5990	51,420
Number of markers	1279	4311
Species	Wheat	Soy

Unbalancedness

- REML implementations (ASREML, REMLF90, AIREMLF90) were not suitable to estimate covariance components without overlapping individuals

- Thus, REML was not used in the unbalanced scenario

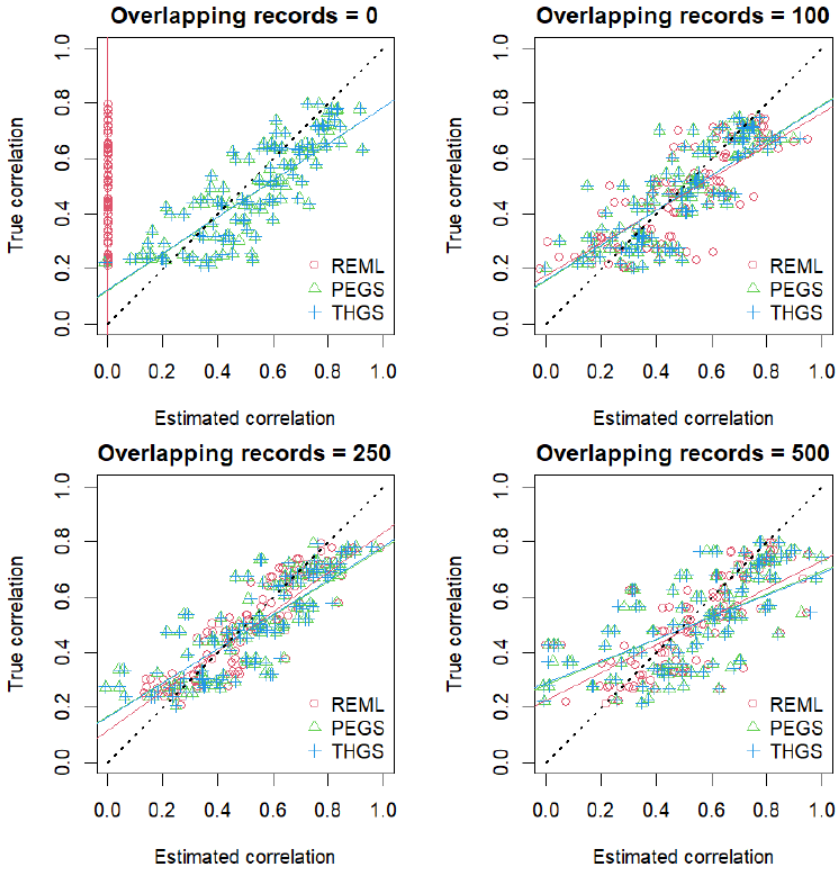


Figure 1: Scatter plot between true and estimated genetic correlations using the soybean dataset with varying number of overlapping individuals across environments.

Elapsed time in small balanced dataset

Method	Time (seconds)
ASREML 4.2	272.6
AIREMLF90	109.8
GIBB3F90	559.8
PEGS, THGS	0.27
Univariate THGS	0.23

Wheat dataset: 10 traits, 599 individuals, 1299 markers
(data available in the BGLR package)

Elapsed time in large unbalanced dataset

Elapsed time to fit multivariate PEGS, THGS

# Markers	# Traits	Time (minutes)
4,311	10	0.2
4,311	50	3.5
4,311	200	80.5
42,034	10	0.8
42,034	50	9.9
42,034	200	123

Soybean dataset: 4628 Individuals
(data available in the SoyNAM package)

More data = less bias = more precision

Table 5 Accuracy of GEBV, regression of TBV on GEBV (Slope), and bias and standard error (SE) of estimates of heritabilities ($\hat{h}^2$) and genetic correlations (GC) with increasing numbers of observations per environment (Obs/Env) in scenario 3, based on 100 replicates of the simulation

Method	Obs/Env	Accuracy	Slope	Bias of $\hat{h}^2$	SE of $\hat{h}^2$	Bias of GC	SE of GC
PEGS	250	0.82 (0.03)	0.98 (0.03)	−0.01 (0.03)	0.07 (0.01)	−0.01 (0.06)	0.17 (0.02)
PEGS	3000	0.96 (0.03)	1.00 (0.03)	−0.01 (0.03)	0.04 (0.01)	0.00 (0.06)	0.13 (0.02)
THGS	250	0.82 (0.03)	0.98 (0.04)	0.00 (0.03)	0.07 (0.01)	−0.02 (0.06)	0.17 (0.02)
THGS	3000	0.96 (0.03)	1.00 (0.03)	−0.01 (0.03)	0.04 (0.01)	0.00 (0.06)	0.13 (0.02)
UV-THGS	250	0.79 (0.03)	1.04 (0.03)	−0.01 (0.03)	0.07 (0.01)	–	–
UV-THGS	3000	0.95 (0.03)	1.00 (0.04)	−0.01 (0.03)	0.04 (0.01)	–	–

Standard errors of statistics are in parenthesis

PEGS pseudo expectation Gauss–Seidel, *THGS* tilde-hat Gauss–Seidel, *UV-THGS* univariate-tilde-hat Gauss–Seidel

1. **Introduction**
   - Rationale and statistical model
2. **Coefficients**
   - Univariate
   - Multivariate
3. **Variances**
   - Univariate
   - Multivariate
4. **Simulations**
   - Elapsed times
   - Benchmarks
5. **Megavariate**
   - Framework
   - Benchmarks
6. **Conclusion**

Megavariate solvers

Key elements

- Scalable for number of response variables
- Covariance components not estimated explicitly

Statistical framework

- Latent spaces for managing dimensionality
- Key tricks: Structural Equations (SEM), Factor analytics (XFA)

Models: MegaLMM (2021), MegaSEM (2024), Canonical Transformation (CT) (1980's)

Structure equation models

SEM: Traits as function of other traits ($\mathbf{y}$ in both sides of the equation)

$$\mathbf{y}_k = \mu_k + \mathbf{Y}_{-k}\lambda_k + \mathbf{Z}_k\boldsymbol{\beta}_k + \mathbf{e}_k$$

For large $\mathbf{Y}$,

(1) regularize, (2) variable selection, (3) dimensionality reduction

MegaLMM

MegaLMM: Mega-scale linear mixed models for genomic predictions with thousands of traits

Daniel E. Runcie[1*], Jiayi Qu[1], Hao Cheng[1] and Lorin Crawford[2]

Shared information

Trait specific information

Step 1: iteratively solve for $\mathbf{F}, \mathbf{\Lambda}, \mathbf{J}$

$$\mathbf{Y} = \mathbf{F\Lambda} + \mathbf{J}$$

$$\mathbf{J} = \mathbf{Z_2 B_2} + \mathbf{E}$$

Step 2: fit model for each latent space

$$\mathbf{F} = \mathbf{Z_1 B_1} + \mathbf{e}$$

Step 3: recover multivariate marker effects

$$\mathbf{B} = \mathbf{B_1 V \Lambda} + \mathbf{B_2}$$

MegaSEM

Step 1: univariate by trait

$$y_k = \mu_k + \mathbf{Z}_k \boldsymbol{\beta}_k^{UV} + e_k$$

Step 2: fit and decompose GEBV matrix
(create latent spaces)

$$\mathbf{G}^{UV} = \mathbf{ZB}^{UV}$$
$$\mathbf{G}^{UV} = \mathbf{UDV}'$$
$$= \mathbf{FV}'$$

Shared information

Trait specific information

Step 3: refit each using latent spaces

$$y_k = \mu_k + \mathbf{F}_k \boldsymbol{\lambda}_k + Z_k \beta_k + e_k$$

Step 4: recover multivariate marker effects

$$\mathbf{B} = \mathbf{B}^{UV}\mathbf{V}\boldsymbol{\Lambda} + B_k$$

MegaLMM vs MegaSEM

	MegaLMM	MegaSEM	CT
Latent spaces	**Stochastic**	**SVD of GEBVs**	**SVD**
Correlated genetics	**YES**	**YES**	**YES**
Correlated residuals	**YES**	**NO**	**YES**
Solver	**BGS**	**PEGS, THGS**	**REML, BGS**
Tunning parameters	**YES**	**NO**	**NO**
Bad scalability	**# Entries**	**# Markers**	**Missing data**

Runtime benchmark

Table 1 Average runtime in minutes (s.e.) for the balanced experimental design based on 10 simulated replicates. Six scenarios vary in terms of the number of environments and individuals (No. environments / No. individuals). Models are ordered based on computational performance. Standard error shown in parenthesis.

Model	Solver	10 / 500	10 / 2,000	50 / 2,000	200 / 2,000	2,000 / 2,000	200 / 20,000
GREML	REML	46.75 (0.37)	172.61 (17.93)	-	-	-	-
D-GREML	REML	0.06 (<0.1)	0.19 (<0.1)	8.32 (3.51)	-	-	-
MegaLMM	MCMC	0.31 (0.01)	4.38 (0.06)	7.23 (1.19)	17.71 (4.02)	130.77 (11.51)	-
MegaSEM	PEGS	<0.01 (<0.01)	0.01 (<0.01)	0.04 (<0.01)	0.14 (<0.01)	2.92 (0.02)	5.26 (0.07)
MV	PEGS	<0.01 (<0.01)	<0.1 (<0.01)	0.02 (<0.01)	9.12 (1.62)	97.14 (1.29)	82.22 (5.71)
XFA	PEGS	<0.01 (<0.01)	<0.1 (<0.01)	0.03 (<0.01)	0.49 (0.09)	-	81.46 (1.38)
HCS	PEGS	<0.01 (<0.01)	<0.01 (<0.01)	0.02 (<0.01)	0.22 (0.04)	38.74 (3.60)	37.74 (4.45)
SCT	PEGS	<0.01 (<0.01)	0.01 (<0.01)	0.04 (<0.01)	0.15 (0.01)	1.65 (0.01)	5.25 (0.05)
UV	PEGS	<0.01 (<0.01)	0.01 (<0.01)	0.04 (<0.01)	0.14 (<0.01)	1.44 (0.01)	5.20 (0.06)

PEGS

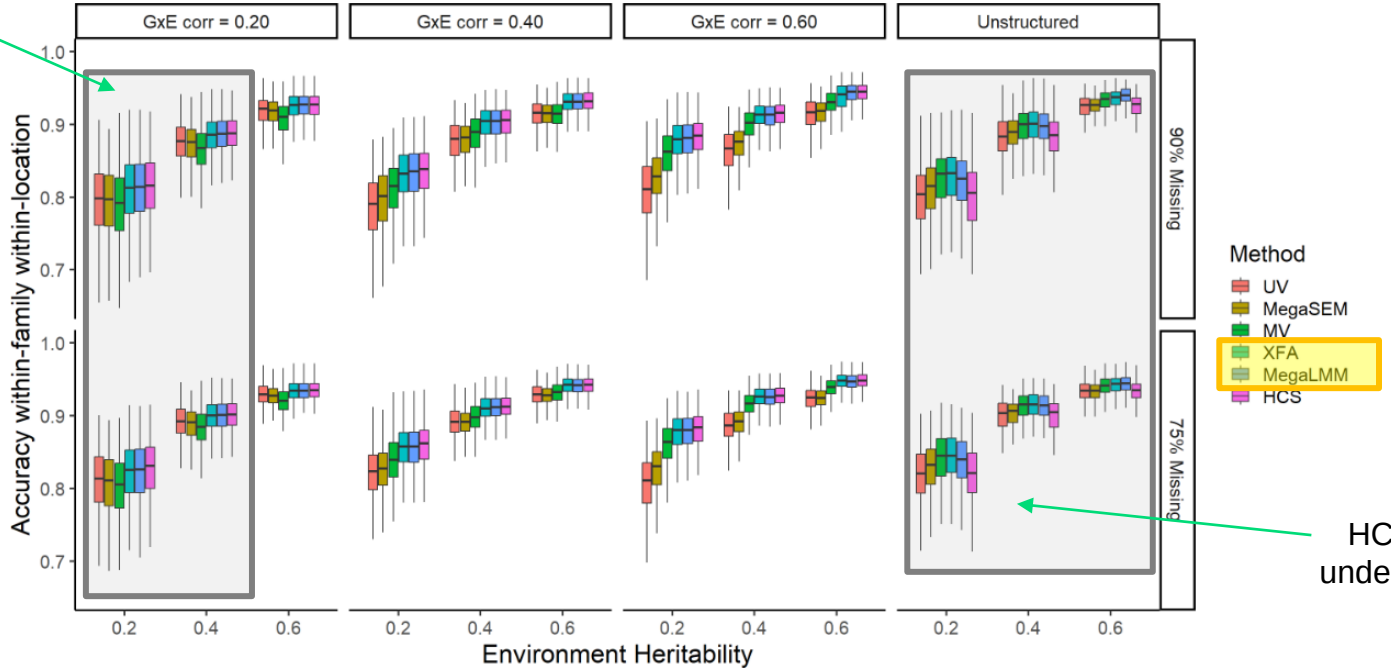
Accuracy benchmark

Table 2 Within environment accuracy for the balanced experimental design based on 10 simulated replicates. Six scenarios vary in terms of the number of environments and individuals (No. environments / No. individuals). Models are ordered based on computational performance. Standard error shown in parenthesis.

Model	Solver	10 / 500	10 / 2,000	50 / 2,000	200 / 2,000	2,000 / 2,000	200 / 20,000
GREML	REML	0.81 (0.03)	0.89 (<0.01)	-	-	-	-
MegaLMM	MCMC	0.78 (0.04)	0.87 (<0.01)	0.87 (<0.01)	0.89 (<0.01)	0.90 (<0.01)	-
MegaSEM	PEGS	0.79 (0.04)	0.88 (<0.01)	0.89 (<0.01)	0.89 (<0.01)	0.89 (<0.01)	0.96 (<0.01)
MV	PEGS	0.81 (0.03)	0.89 (<0.01)	0.89 (<0.01)	0.90 (<0.01)	0.88 (<0.01)	0.96 (<0.01)
XFA	PEGS	0.80 (0.04)	0.89 (<0.01)	0.89 (<0.01)	0.89 (<0.01)	-	0.96 (<0.01)
HCS	PEGS	0.81 (0.03)	0.88 (<0.01)	0.88 (<0.01)	0.88 (<0.01)	0.88 (<0.01)	0.96 (<0.01)
SCT	PEGS	0.81 (0.03)	0.89 (<0.01)	0.88 (<0.01)	0.87 (<0.01)	0.87 (<0.01)	0.95 (<0.01)
UV	PEGS	0.78 (0.04)	0.87 (<0.01)	0.87 (<0.01)	0.87 (<0.01)	0.87 (<0.01)	0.95 (<0.01)

Sparse testing

MV suffers under low GxE and low H2



HCS suffers under UNS GxE

Figure 1 Prediction accuracy within-family and within-environment using 100 simulated environments with varying heritability, percentage of missing values, and GxE correlation. Genomic information was sourced from nine soybean bi-parental families.

Genomes-to-Fields dataset

Table 3 Predictive ability from the 2022 G2F GxE prediction competition. Corn grain yield observed in 4,836 hybrids across 217 locations (2014-2021) predicting 548 hybrids observed across 21 environments (2022). Models are ordered based on the pairwise metric. Standard error shown in parenthesis.

Model	Pairwise	Region	Overall
UVW	0.08 (0.03)	0.22 (0.14)	0.27 (0.11)
MV	0.12 (0.05)	0.27 (0.12)	0.30 (0.11)
MegaSEM	0.13 (0.05)	0.25 (0.15)	0.27 (0.11)
MegaLMM	0.18 (0.06)	0.24 (0.19)	0.27 (0.10)
XFA	0.21 (0.07)	0.31 (0.13)	0.35 (0.12)
HCS	0.24 (0.09)	0.34 (0.11)	0.36 (0.11)
UVA	-	-	0.35 (0.12)

1. **Introduction**
   - Rationale and statistical model
2. **Coefficients**
   - Univariate
   - Multivariate
3. **Variances**
   - Univariate
   - Multivariate
4. **Simulations**
   - Elapsed time
   - Study 1: Comparison to REML in small balanced data
   - Study 2: Performance in large unbalanced data
   - Megavariate extension
5. **Conclusion**

General recommendation

- REML for balanced sets, small datasets with few traits, or pairwise covariance estimations
- Bayesian Gibbs Sampling for 5-20 traits, small-to-moderate size datasets
- PEGS, THGS for 1-100 traits, mid-to-large datasets
- Specialized methods for 100+ traits - MegaLMM, MegaSEM

Thank you for your attention!

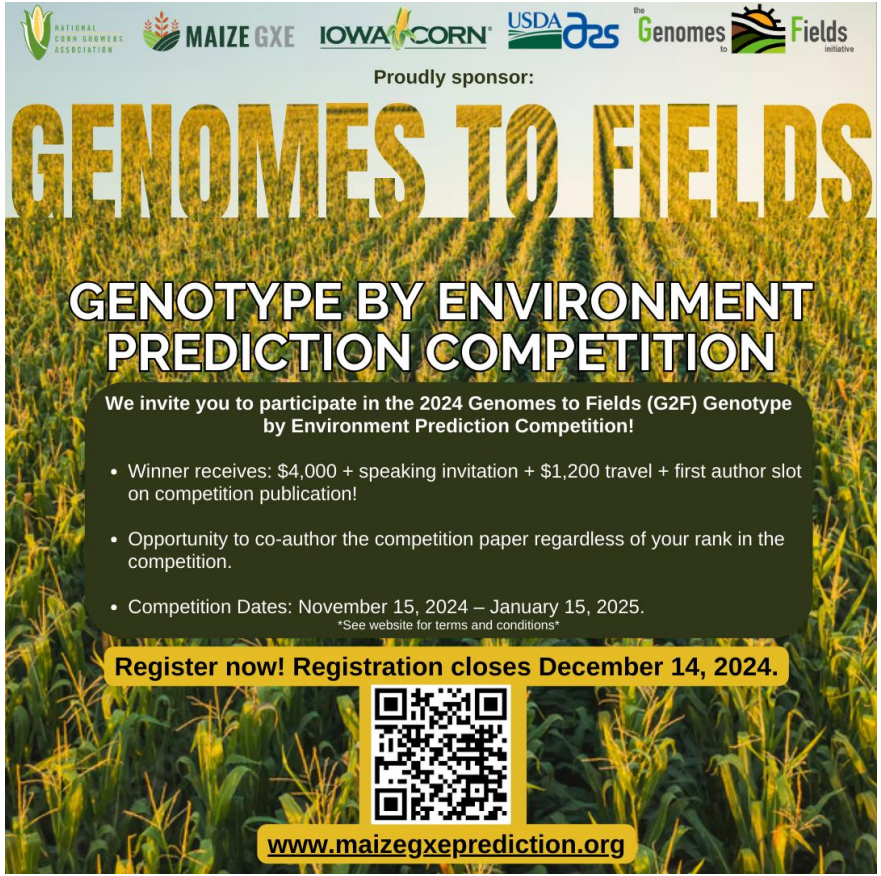
Final remarks:

1) Traditional multivariate models are valuable, but computationally unfeasible
2) Efficient estimation of coefficients and variances enable large multivariate models
3) Latent space models enable even larger dimensionalities

Questions??

Alencar Xavier

Alencar.Xavier@Corteva.com


NATIONAL CORN GROWERS ASSOCIATION
MAIZE GXE
IOWA CORN
USDA ARS
the Genomes to Fields initiative
Proudly sponsor:
GENOMES TO FIELDS
GENOTYPE BY ENVIRONMENT PREDICTION COMPETITION
We invite you to participate in the 2024 Genomes to Fields (G2F) Genotype by Environment Prediction Competition!
• Winner receives: $4,000 + speaking invitation + $1,200 travel + first author slot on competition publication!
• Opportunity to co-author the competition paper regardless of your rank in the competition.
• Competition Dates: November 15, 2024 – January 15, 2025.
See website for terms and conditions
Register now! Registration closes December 14, 2024.
www.maizegxeprediction.org