

Modern genetic evaluation systems: A Python-based programming approach

Kristin Lee, Gordon Vander Voort, Ricardo Ventura, Flavio Schenkel, Angela Cánovas

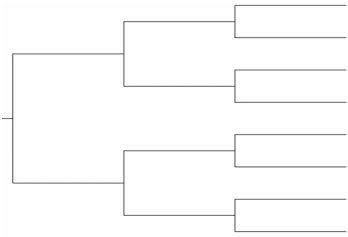
Centre for Genetic Improvement of Livestock, Department of Animal Biosciences, University of Guelph, Guelph, Ontario, Canada



The 75th EAAP Annual Meeting

Florence, Italy · September 1 – 5, 2024

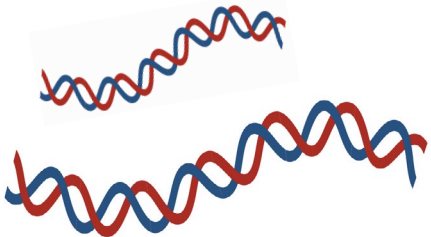
Genetic evaluation systems



Pedigree



Phenotypes



Genotypes

+

$$Ax = b$$

Linear
equation

```
scanId = cellfun(@(c) ~isempty(c), scanId);  
scanId = tmp(scanId);  
  
nScans = numel(scanId(:));  
  
scans = struct('Day', string(1:nScans), 'ID', string(1:nScans),  
             'Labels', cell(nScans, 1), 'Importance', zeros(nScans, 1),  
             'ScanLabel', false, 'Pos', cell(nScans, 1),  
             'File', cell(nScans, 1), 'Range', cell(nScans, 1));  
  
if nargin > 1  
    progressBar = ProgressBar(nScans, 'Scanning');  
end  
  
nScans = 1;  
for i = 1:nScans  
    for j = 1:10  
        scanId = scanId(i,j);  
    end  
end
```

Numerical
algorithm



Estimated
breeding values

Traditional genetic evaluation systems are developed in C or Fortran

Benefits

- Lower-level languages
- Efficiency
- Numerical & scientific computing
- Performance optimization
- Explicit memory management



Limitations

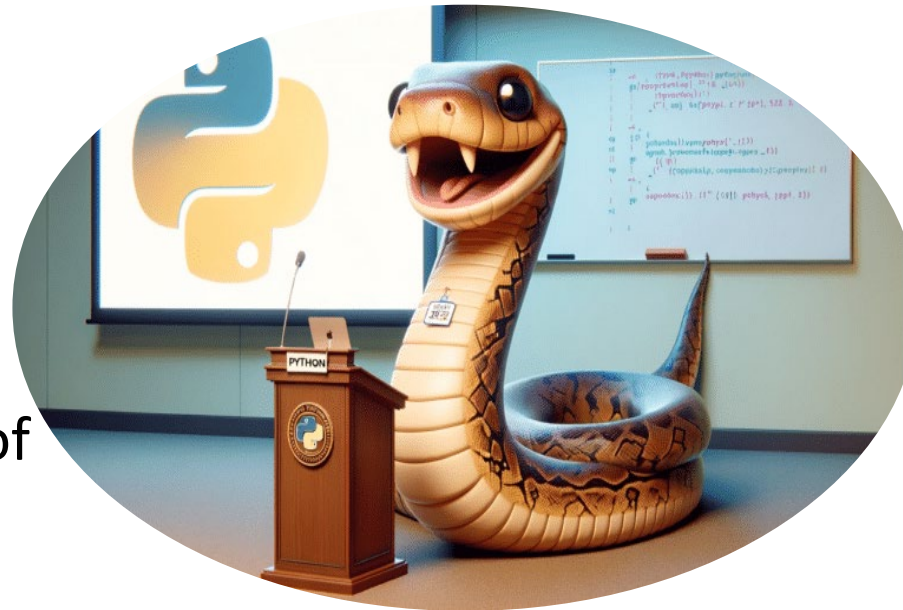
- Less flexibility
 - Slower development
 - Higher maintenance
- High costs
- Affect innovation

Python could be used to develop an adaptable software that can produce accurate breeding values within practical timeframes

Benefits

- Higher-level language
- Faster development
- Easier maintenance

→ Easy & fast integration of new features



Limitations

- Slower performance
- Higher memory requirements

→ Modern computers

Data

Simulated purebred angus beef cattle:

- AlphaSimR
- Sample size: $N = 976,400$
- Pedigree: 15 generations
- Phenotypes: Birth weight, weaning gain, post-weaning gain
- Masked phenotypes for final simulated generation



Data

Genotypes:

- 48,981 SNP markers
- Random sampling from final simulated generation
- Sample Size: 10,000
- 3 replicates of sample size



Linear model

$$\begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \mathbf{y}_3 \end{bmatrix} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \end{bmatrix} + \begin{bmatrix} \mathbf{Z}_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{Z}_2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{Z}_3 \end{bmatrix} \begin{bmatrix} \mathbf{a}_1 \\ \mathbf{a}_2 \\ \mathbf{a}_3 \end{bmatrix} + \begin{bmatrix} \mathbf{e}_1 \\ \mathbf{e}_2 \\ \mathbf{e}_3 \end{bmatrix}$$

$\mathbf{y}$ = vector of phenotype records

μ = phenotypic mean

$\mathbf{Z}$ = incidence matrix relating phenotypes to animal effects

$\mathbf{a}$ = vector of random additive genetic effects

$\mathbf{e}$ = vector of residual effects

Assumptions

$$\text{Var} \begin{bmatrix} \mathbf{a} \\ \mathbf{e} \end{bmatrix} = \begin{bmatrix} \mathbf{G} \otimes \mathbf{Rel} & \mathbf{0} \\ \mathbf{0} & \mathbf{R} \otimes \mathbf{I} \end{bmatrix}$$

$\mathbf{Rel}$ = relationship matrix

$\mathbf{G}$ = (co)variance matrices for genetic effects

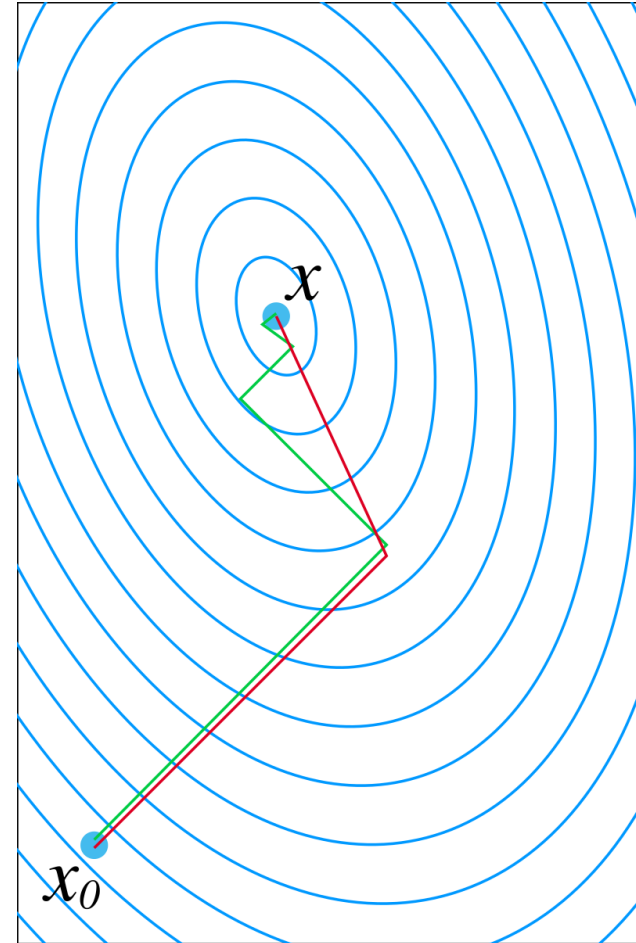
$\mathbf{R}$ = (co)variance matrices for residual effects

$\mathbf{I}$ = identity matrix

Numerical algorithm

Preconditioned conjugate gradient with iteration on data:

- Diagonal preconditioner
- Serial, multiprocessing, sparse
- Convergence criteria = $\frac{\|\mathbf{r}_i\|}{\|\mathbf{b}\|} = 1 \times 10^{-5}$



Software performance

Runtime analysis:

- The software was executed five times
- Average time elapsed of major components was reported



Server specifications:

- Operating system: Linux (64-bit)
- Memory (RAM): 96GB
- Processor: Intel(R) Xeon(R) Gold 6242 CPU @ 2.80GHz
- Cores: 8



Results verification

MiXBLUP:

- Commercial software
- Genetic evaluation system
- Pearson correlation
- Estimated breeding values



Software performance

Average runtime for key components of the Python genetic evaluation system

Component		Time (seconds)
Read data		1.79
Preprocessing		38.51
Relationship matrix		
	Mendelian sampling	0.19
	Inbreeding	15,020.00
	Inverse of the relationship matrix	3.57
Preconditioned conjugate gradient		
	A) Serial processing	2,688.15
	B) Multiprocessing	933.24
	C) Sparse matrix operations	770.12
Write results		4.09

Software performance

Average runtime for key components of the Python genetic evaluation system

Component	Time (seconds)
Read data	1.79
Preprocessing	38.51
Relationship matrix	
Mendelian sampling	0.19
Inbreeding	15,020.00
Inverse of the relationship matrix	3.57
Preconditioned conjugate gradient	
A) Serial processing	2,688.15
B) Multiprocessing	933.24
C) Sparse matrix operations	770.12
Write results	4.09

Software performance

Average runtime for key components of the Python genetic evaluation system

Component		Time (seconds)	
Read data		1.79	
Preprocessing		38.51	
Relationship matrix			
	Mendelian sampling	0.19	
	Inbreeding	15,020.00	4 hours 10 minutes
	Inverse of the relationship matrix	3.57	
Preconditioned conjugate gradient			
	A) Serial processing	2,688.15	
	B) Multiprocessing	933.24	
	C) Sparse matrix operations	770.12	
Write results		4.09	

Software performance

Average runtime for key components of the Python genetic evaluation system

Component		Time (seconds)	
Read data		1.79	
Preprocessing		38.51	
Relationship matrix			
Mendelian sampling		0.19	
Inbreeding		15,020.00	
Inverse of the relationship matrix		3.57	
Preconditioned conjugate gradient			
	A) Serial processing	2,688.15	44 minutes 48 seconds
	B) Multiprocessing	933.24	15 minutes 33 seconds
	C) Sparse matrix operations	770.12	12 minutes 50 seconds
Write results		4.09	

Software performance

Total runtime for the MiXBLUP genetic evaluation:

- 58 seconds



Results verification

Pearson correlation coefficient (r) between estimated breeding values (EBV) of the Python and MiXBLUP genetic evaluation systems

Trait	$r_{\text{EBV:EBV}}$
Birth weight	1.0
Weaning gain	1.0
Post-weaning gain	1.0

Software performance

Average runtime for key components of the Python genetic evaluation system

Component		Time	
Read genotypes		56.06	
Preprocessing		17.95	
Relationship matrix			
	Genomic relationship matrix (G)	29.49	
	Blend & tune G	5.06	
	Invert G	10.11	
	Construct and invert A22	52,040.54	
	Construct H-1	31.51	
Preconditioned conjugate gradient			
	Sparse matrix operations	2,367.50	

Software performance

Average runtime for key components of the Python genetic evaluation system

Component	Time
Read genotypes	56.06
Preprocessing	17.95
Relationship matrix	
Genomic relationship matrix (G)	29.49
Blend & tune G	5.06
Invert G	10.11
Construct and invert A22	52,040.54
Construct H-1	31.51
Preconditioned conjugate gradient	
Sparse matrix operations	2,367.50

Software performance

Average runtime for key components of the Python genetic evaluation system

Component		Time	
Read genotypes		56.06	
Preprocessing		17.95	
Relationship matrix			
	Genomic relationship matrix (G)	29.49	
	Blend & tune G	5.06	
	Invert G	10.11	
Construct and invert A22		52,040.54	
Construct H-1		31.51	
Preconditioned conjugate gradient			
Sparse matrix operations		2,367.50	

Software performance

Average runtime for key components of the Python genetic evaluation system

Component	Time	
Read genotypes	56.06	
Preprocessing	17.95	
Relationship matrix		
Genomic relationship matrix (G)	29.49	
Blend & tune G	5.06	
Invert G	10.11	
Construct and invert A22	52,040.54	14 hours 27 minutes
Construct H-1	31.51	
Preconditioned conjugate gradient		
Sparse matrix operations	2,367.50	

Software performance

Average runtime for key components of the Python genetic evaluation system

Component	Time
Read genotypes	56.06
Preprocessing	17.95
Relationship matrix	
Genomic relationship matrix (G)	29.49
Blend & tune G	5.06
Invert G	10.11
Construct and invert A22	52,040.54
Construct H-1	31.51
Preconditioned conjugate gradient	
Sparse matrix operations	2,367.50

Software performance

Average runtime for key components of the Python genetic evaluation system

Component		Time	
Read genotypes		56.06	
Preprocessing		17.95	
Relationship matrix			
Genomic relationship matrix (G)		29.49	
Blend & tune G		5.06	
Invert G		10.11	
Construct and invert A22		52,040.54	
Construct H-1		31.51	
Preconditioned conjugate gradient			
	Sparse matrix operations	2,367.50	39 minutes 27 seconds

Software performance

Total runtime for the MiXBLUP genetic evaluation:

- 15 minutes 14 seconds



Results verification

Pearson correlation coefficient (r) between estimated breeding values for genotyped individuals of the Python & MiXBLUP genetic evaluation systems

Trait	$r_{\text{EBV:EBV}}$
Birth weight	0.99
Weaning gain	0.99
Post-weaning gain	0.99



Python for genetic evaluation system development:

- Simple development process
- Accurate
- Practical timeframes for small to medium sized breeding programs
- Components needing optimization identified

Acknowledgements



Contact:

Kristin Lee

klee32@uoguelph.ca

UNIVERSITY
of GUELPH



AgSights

Data. Knowledge. Insight.

Mitacs