PX × AI : algorithmics for better convergence in restricted maximum likelihood estimation

Karin Meyer

Animal Genetics and Breeding Unit, University of New England,
Armidale, NSW 2351, Australia

kmeyer@didgeridoo.une.edu.au

Outline

REML estimation I

- Widely used for estimation of variance components
- Represents a constrained optimisation problem
- Iterative schemes to maximise $\log \mathcal{L}$ differ in
  - Convergence rate
  - Computational demands
  - Ease of implementation
  - Reliability
- 'Average information' (AI) algorithm preferred
  - "Often converges in a few rounds" (Thompson *et al.*, 2005)
  - Anecdotal evidence of convergence problems
    - bad starting values
    - multiple parameters or random effects
    - at boundary of parameter space

REML estimation II

- 'Expectation-Maximisation' (EM) $\longrightarrow$ alternative
  - Stable
    - monotone increase in $\log \mathcal{L}$
    - estimates within parameter space
  - Painfully slow to converge
- New 'fast' EM type algorithms $\longrightarrow$ 1990 onwards
  - 'Parameter Expanded' (PX) EM (Liu *et al.*, 1998)
    - applicable to REML mixed model analyses
    - much faster than standard EM (Foulley and van Dyk, 2000)
    - no 'real' application so far
    - only limited comparison with AI (Cullis *et al.*, 2004)

Objectives

This study

- Examine convergence rates of PX-EM algorithm
- Compare PX-EM with AI algorithm
  - Beef cattle examples
  - Problem analyses

Newton methods for optimisation

Why AI REML can fail

- AI algorithm uses 2nd derivatives of $\log \mathcal{L}$ (data part)
  - unconstrained $\rightarrow$ reparameterisation
  - $\log \mathcal{L}$ not guaranteed to increase
- Modified Newton method

  $\boldsymbol{\theta}^{t+1} = \boldsymbol{\theta}^t + \omega \boldsymbol{\Delta}$ with $\boldsymbol{\Delta} = -(\mathbf{H}^t)^{-1}\mathbf{g}^t$
  - Phase I : Damped $\rightarrow 0 < \omega < 1$
    - 'far' from maximum of $\log \mathcal{L}$
    - estimate ω using approximation of $\log \mathcal{L}$
  - Phase II : Quadratically convergent $\rightarrow \omega = 1$
    - close to maximum of $\log \mathcal{L}$
    - norm of $\mathbf{g}$ converges quadratically to zero
    - few iterates
  - Ensure $\mathbf{H}$ is 'safely' positive definite $\rightarrow$ add $\lambda \mathbf{I}$

'Expectation Maximisation' algorithms

- Principle of EM
  - E-step : Take expectation of *complete* likelihood $\mathcal{Q}$
    - impute 'missing values'
    - assume current parameter values maximise likelihood
  - M-step : Equate 1st derivatives of $\mathcal{Q}$ to zero
- Standard EM

  $\mathbf{y} = \mathbf{Xb} + \mathbf{Zu} + \mathbf{e}$ with $\text{Var}(\mathbf{u}) = \boldsymbol{\Sigma} \otimes \mathbf{A}$
  - Estimate $\boldsymbol{\Sigma}$
- PX - EM

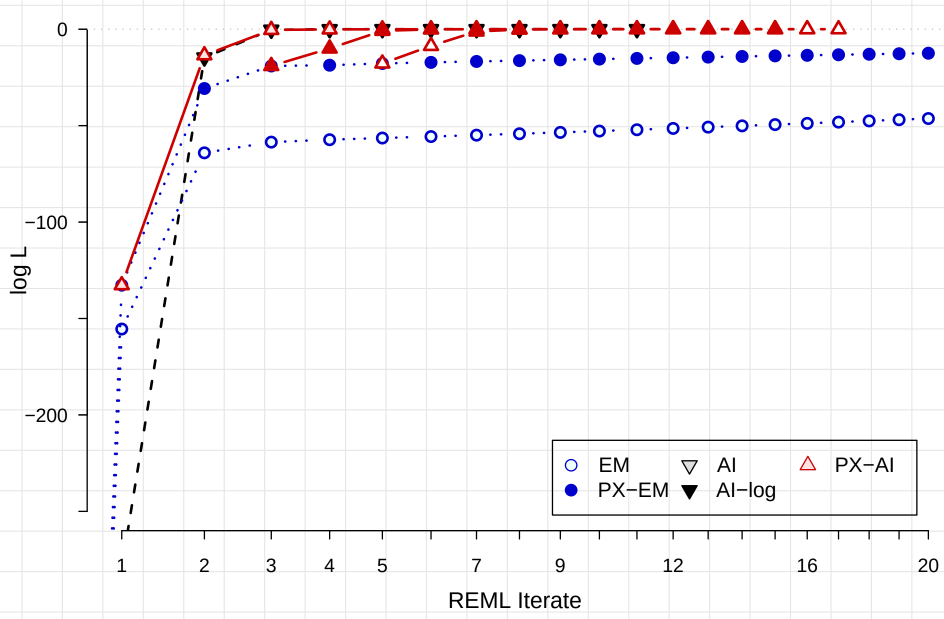
  $\mathbf{y} = \mathbf{Xb} + \mathbf{Z}(\boldsymbol{\alpha} \otimes \mathbf{I})\mathbf{u}^* + \mathbf{e}$ with $\text{Var}(\mathbf{u}^*) = \boldsymbol{\Sigma}^* \otimes \mathbf{A}$
  - Estimate $\boldsymbol{\Sigma}^*$ and $\boldsymbol{\alpha}$
  - Reduce to $\boldsymbol{\Sigma} = \boldsymbol{\alpha}'\boldsymbol{\Sigma}^*\boldsymbol{\alpha}$

    $\Rightarrow$ Partially account for error in $\mathcal{Q}$ due to $\boldsymbol{\theta}^t \neq \hat{\boldsymbol{\theta}}_{max}$

Example 1

- Multivariate analysis : 3 traits
  - Birth, weaning & yearling weight of beef cattle
- 3 random effects
  - Direct, additive genetic
  - Maternal, additive genetic
  - Maternal, permanent environmental
- 24 parameters to be estimated
  - assume direct-maternal genetic covariances are zero
- 10 479 records on 4 953 animals; 6 247 animals in pedigree

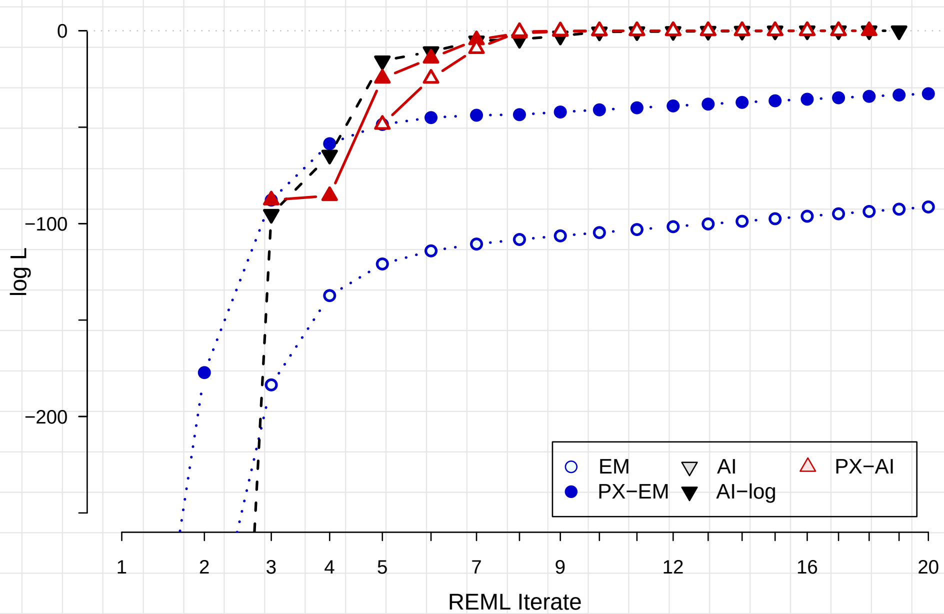
‘Good’ starting values



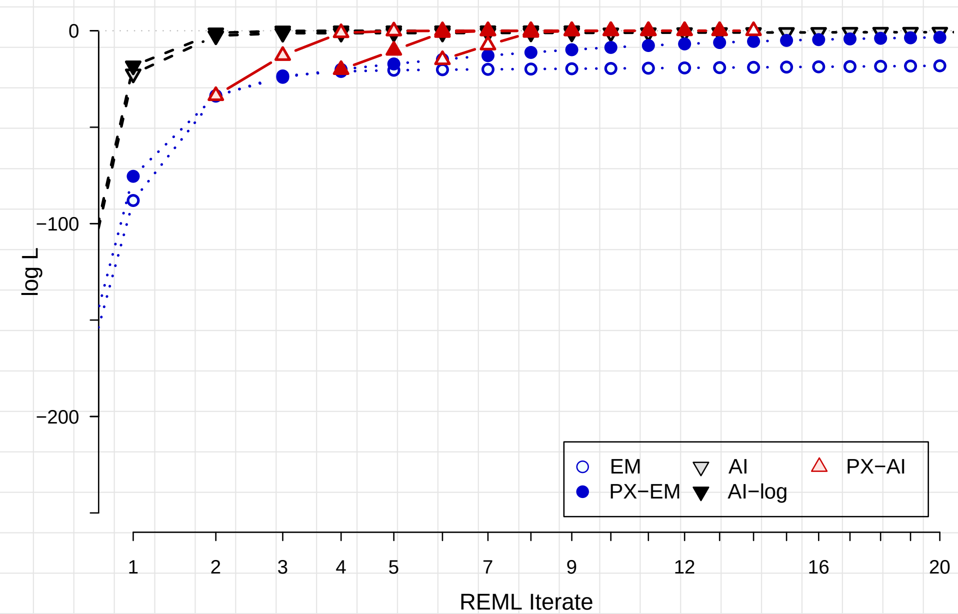
Example 2

- Multivariate analysis : 8 traits
  - 4 ‘carcass’ traits measured by live ultra-sound scanning
  - records on heifers/steers and bulls treated as different traits
- Simple animal model
- 56 parameters to be estimated
  - 36 genetic covariances
  - 20 residual covariances
- Records from a single, large herd
  - 20 171 records on 5 605 animals
  - 8 044 animals in pedigree

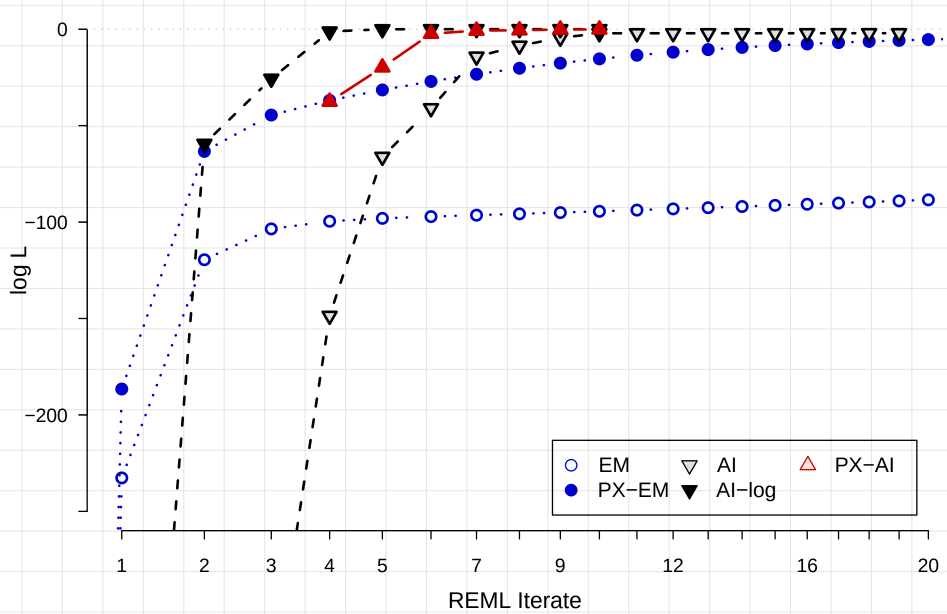
‘Bad’ starting values



‘Good’ starting values



'Bad' starting values



Conclusions

- PX-EM algorithm very useful in early iterates
- PX $\times$ AI algorithm
  - safe-guard against bad starting values
  - eliminate ambiguities of modifying AI algorithm
- Disadvantages
  - extra programming effort
  - requires inverse of coefficient matrix
- AI REML performs extremely well
  - with good starting values
  - when properly controlled
- **Get good starting values !**

Summary of results

- Similar results for other cases
- PX-EM
  - excellent performance in early iterates
  - most advantageous for
    - analyses with 'bad' starting values
    - models with several random effects
    - estimates at boundary of parameter space
  - convergence rates as for standard EM in later iterates
- AI algorithm
  - can be 'temperamental' in early iterates
  - rapid convergence once close to maximum of log $\mathcal{L}$
  - need stringent control to ensure reliable convergence
    - choices not always clear-cut