

Soluções Computacionais para a Predição de Valores Genéticos em Animais de Produção

Manual da Versão 1.3 do Programa Intergeren



*Empresa Brasileira de Pesquisa Agropecuária
Embrapa Pecuária Sul
Ministério da Agricultura, Pecuária e Abastecimento*

DOCUMENTOS 153

Soluções computacionais para a predição de valores genéticos em animais de produção

Manual da versão 1.3 do programa Intergen

*Fernando Flores Cardoso
Vinicius Silva Junqueira
Rodrigo Reis Mota*

Exemplares desta publicação podem ser adquiridos na:

Embrapa Pecuária Sul
Rodovia BR-153, Km 632,9
Vila Industrial, Zona Rural, C. Postal 242
CEP 96401-970, Bagé, RS
Fone: +55 (53) 3240-4650
Fax: +55 (53) 3240-4651
www.embrapa.br/fale-conosco/sac

Comitê Local de Publicações
da Embrapa Pecuária Sul

Presidente
Fernando Flores Cardoso

Secretária-Executiva
Márcia Cristina Teixeira da Silveira

Membros
*Elisa Köhler Osmari, Estefanía Damboriarena,
Fabiane Pinto Lamego, Graciela Olivella Oliveira,
Jorge Luiz Sant'Anna dos Santos, Lisiane Brisolara,
Robert Domingues, Sérgio de Oliveira Jüchem*
Suplentes
Henry Gomes de Carvalho, Marcos Jun Iti Yokoo

Supervisão editorial
Lisiane Brisolara

Revisão de texto
Fernando Goss

Normalização bibliográfica
Graciela Olivella Oliveira

Projeto gráfico da coleção
Carlos Eduardo Felice Barbeiro

Editoração eletrônica
*Ana Tailise Estevão
Luana Noble de Oliveira*

Foto da capa
Felipe Santos da Rosa

1ª edição
Publicação digitalizada (2019)

Todos os direitos reservados.

A reprodução não autorizada desta publicação, no todo ou em parte,
constitui violação dos direitos autorais (Lei nº 9.610).

Dados Internacionais de Catalogação na Publicação (CIP)
Embrapa Pecuária Sul

Cardoso, Fernando Flores

Soluções computacionais para a predição de valores genéticos em animais
de produção: manual da versão 1.3 do programa Intergen / Fernando Flores Car-
doso, Vinicius Silva Junqueira, Rodrigo Reis Mota. — Bagé : Embrapa Pecuária
Sul, 2019.

PDF (41 p.) : 21 cm x 29,7 cm. — (Documentos / Embrapa Pecuária Sul, ISSN
1982-5390; 153)

1. Genética animal. 2. Produto animal. 3. Programa de computador. I. Vinicius
Silva. II. Mota, Rodrigo Reis. III. Título. IV. Série.

CDD 636.0821

Autores

Fernando Flores Cardoso

Médico Veterinário, Doutor em Ciência Animal com ênfase em Genética e Melhoramento Animal, pesquisador da Embrapa Pecuária Sul, Bagé, RS

Vinicius Silva Junqueira

Médico Veterinário, Doutor em Genética e Melhoramento, Uberlândia, MG

Rodrigo Reis Mota

Zootecnista, Doutor em Genética e Melhoramento, pesquisador assistente da Universidade de Liège - Gembloux Agro-Bio Tech (GxABT), Gembloux, Bélgica

Apresentação

O investimento em pesquisa, desenvolvimento e inovação (PD&I) é fundamental para a competitividade das cadeias produtivas. Dentro os temas prioritários para a geração de produtos e tecnologias através da pesquisa científica na Embrapa Pecuária Sul esta o melhoramento genético animal. A disseminação dos conhecimentos gerados nessa área possibilita que os beneficiários do processo, ou seja, produtores, técnicos, estudantes e demais interessados nas novas tecnologias genéticas, possam selecionar e acasalar animais mais produtivos e eficientes, que venham a melhor o resultado econômico nos sistemas de produção.

Neste contexto, o Programa Intergen versão 1.3 foi desenvolvido pela Embrapa Pecuária Sul para disponibilizar aos programas de melhoramento animal uma ferramenta para predição do mérito genético de animais de produção, que ao mesmo tempo é flexível para lidar com diferentes modelos e estruturas de população e de alta capacidade para analisar grandes volumes de dados.

Nesta publicação da Série de Transferência de Tecnologia da Embrapa é descrito ao usuário como implementar todas as capacidades do programa tanto na pesquisa científica quanto na rotina das avaliações genéticas.

Esperamos que o emprego dos conhecimentos disponibilizados nesta obra nos programas de avaliação genética possa colaborar com o aumento do progresso genético dos rebanhos e que a genética comprovadamente superior se dissemine para permear suas vantagens na atividade pecuária do sul do Brasil.

Fernando Flores Cardoso
Chefe Adjunto de Pesquisa

Sumário

Introdução	9
O que há de novo na versão 1.3	9
Arquivo de parâmetros	10
Como rodar o programa	23
Como continuar uma cadeia já iniciada.....	24
Arquivos de saída do programa	24
Referências	25
Anexos.....	26
A.1. Modelo animal com prioris próprias	26
A.2. Modelo animal utilizando a inversa da matriz de parentesco genômico.....	27
A.3. Modelo animal usando uma matriz do numerador do parentesco médio de Henderson	28
A.4. Modelo animal com grupos genéticos de Westell.....	30
A.5. Modelo de incerteza de paternidade para filhos de acasalamentos com reprodutores múltiplos	31
A.6. Modelo estrutural para variância e robustez a dados extremos	32
A.7. Modelo animal multirracial	33
A.8. Modelo de normas de reação	35
A.9. Modelo de normas de reação utilizando a inversa da matriz de parentesco genômico	36
A.10. Modelo animal via BLUP.....	38
A.11. Modelo animal via BLUP com grupos genéticos de Westell sem modelagem de reprodutores múltiplos	39
A.12. Modelo animal via BLUP com efeito normal prior para efeitos fixos.....	40

Introdução

Este documento tem por objetivo descrever a funcionalidade e utilização da Versão 1.3 do programa INTERGEN, originalmente descrito em Cardoso (2008, 2010). O programa foi desenvolvido na Embrapa Pecuária Sul em linguagem Fortran 90/95 com capacidade de implementar modelos hierárquicos de Bayes e estimar seus parâmetros por meio de métodos Monte Carlo via cadeias de Markov (MCMC) e dos melhores preditores lineares não viesados (BLUP) (Sorensen; Gianola, 2002).

O INTERGEN está disponível na seção de produtos e serviços do portal da Embrapa, onde se deve pesquisar pela palavra chave “intergen”. Os modelos possíveis de serem ajustados no INTERGEN contemplam situações complexas abordadas em estudos de genética quantitativa, utilizando dados de desempenho de animais de produção, incluindo, entre outros:

- Modelo de incerteza de paternidade, na presença de acasalamentos com reprodutores múltiplos;
- Modelo multirracial, para populações cruzadas com múltiplas composições raciais;
- Modelo de normas de reação para estudos de interação genótipo-ambiente;
- Modelo robusto, para conjuntos de registros com dados extremos;
- Modelo estrutural de variâncias, para ajustar heterogeneidade de variância residual.

O software é baseado na família de programas BLUPF90 desenvolvidas por Misztal et al. (2002) e utiliza bibliotecas para geração de números aleatórios e operações com matrizes esparsas, na implementação da metodologia de modelos mistos (Henderson, 1984) e MCMC.

O programa INTERGEN, apesar de permitir modelagem complexa, possui uma interface bastante simples, sendo acionado através de uma janela do Command Prompt do DOS no Windows ou terminal Linux/Mac OS e é controlado por um arquivo de parâmetros (“parameterfile”), o qual contém as informações sobre os arquivos de dados, de pedigree e outros arquivos definidos pelo usuário, sobre os efeitos no modelo e sobre o tipo de análise a ser implementada.

O que há de novo na versão 1.3

- Uma sub-rotina para predição dos valores genéticos via BLUP foi incorporada. O algoritmo é flexível o suficiente para ajustar modelos uni e multi-características, considerando homoscedasticidade de variância residual;
- Uma sub-rotina foi incorporada para permitir a especificação de matrizes de (co)variância definidas pelo usuário, permitindo a implementação da seleção genômica;
- Uma sub-rotina para ponderação do resíduo foi implementada, permitindo que sejam aplicados diferentes pesos para as características;
- Uma sub-rotina foi incorporada para permitir ao usuário a inclusão de uma informação a priori sobre o valor esperado (μ_p) e a variância (V_p) dos efeitos fixos definidos no modelo, efeito *normal prior*.

- Uma sub-rotina para o cálculo da inversa da variância de segregação mendeliana dos animais presentes no arquivo de pedigree foi incluída, possibilitando que o usuário não tenha mais necessidade de informar esse valor no arquivo de pedigree;
- Uma sub-rotina para o cálculo do fenótipo predito ($\tilde{y}=\hat{y}-\hat{e}$), fenótipo ajustado ($\tilde{y}=y-X\hat{b}$) e resíduo ($\hat{e}$) em análises BLUP foi incorporado.

Arquivo de parâmetros

Uma vez acionado, o programa INTERGEN solicita ao usuário a especificação de um arquivo de parâmetros, que deve ser preparado previamente em texto não formatado tipo ASCII.

O arquivo de parâmetros apresenta um formato geral, descrito com detalhes abaixo. As seções são definidas por cabeçalhos em letras maiúsculas (apresentadas em negrito neste documento), que devem aparecer exatamente como descritas e numa única linha do arquivo ASCII – embora aqui possam aparecer em duas linhas por limitação de tamanho de linha no presente documento. Na linha subsequente, os campos que são definidos pelo usuário são listados e campos opcionais aparecem entre parênteses.

Comentários podem ser incluídos no arquivo de parâmetros, desde que se inicie a linha de comentário com o símbolo #. Quando o programa encontra esse símbolo (#), todas as informações posteriores são desconsideradas.

Na versão 1.3 é possível realizar análises MCMC e BLUP, sendo que o tipo de análise deverá ser definido nas linhas iniciais do arquivo de parâmetros (arquivos de exemplos para ambas as análises são disponibilizados nos anexos deste manual).

A análise BLUP é definida ao incluir a primeira linha do arquivo de parâmetros com a seção abaixo:

BLUP: SOLVE_METHOD CONVERGENCE_CRITERIA [STDERR]

método de resolução do sistema de equações, critério de convergência, opção para salvar o desvio padrão do erro de predição

Exemplo

PCG 1e-12 se

Detalhes

Os métodos de resolução do sistema de equações disponíveis são PCG (padrão), SOR ou FSPAK

O critério de convergência padrão (default) é 1e-12.

O parâmetro STDERR é opcional e o padrão é não armazenar tais informações no arquivo solutions. Caso a string se seja identificada, o programa armazena as informações do desvio do erro de predição.

A análise via MCMC é definida ao incluir as linhas iniciais com as seções abaixo:

MCMC_CHAIN: TOTAL_CYCLES BURN_IN THINNING_INTERVAL

número total de ciclos MCMC, período de aquecimento, intervalo para salvar amostras

Exemplo

400000 100000 100

SEED

número inteiro

Detalhes

Semente para iniciar o processo de geração de variáveis aleatórias, o que permite repetir exatamente a mesma cadeia.

Exemplo

123

RESTART: Y/N? [CYCLE_TO_RESTART]

y/n, (ciclo para continuar a cadeia)

Detalhes

Informa se análise é continuidade de cadeia já em andamento. **y** = continuar cadeia anterior (deve informar o ciclo para reiniciar), **n** = iniciar nova.

Exemplo

y 10001

As próximas seções do arquivo de parâmetros são iguais para análises MCMC e BLUP.

DATAFILE NAME N_RECORDS

nome do arquivo de dados, número de linhas de dados

Detalhes

Os arquivos de dados são arquivos texto não formatados (ASCII), com dados para todos os efeitos no modelo em colunas, com pelo menos um espaço em branco entre colunas, e uma linha para cada registro.

No caso de modelo animal, deve conter a identificação (ID) do animal ao qual o registro pertence, de acordo com a codificação no arquivo de pedigree.

Todos os efeitos classificatórios devem ser recodificados de 1 ao número de classes.

Não existe uma ordem necessária dos efeitos no arquivo, entretanto, por questão de organização, recomendamos o seguinte padrão: primeira coluna com ID do animal, a seguir ID da mãe se houver efeito materno, depois listar os demais efeitos classificatórios, seguidos das covariáveis e finalmente das variáveis respostas.

Dados perdidos devem ser codificados com 0.

Exemplo

manimal.dat 22124

NUMBER_OF_TRAITS

número de características respostas na análise

Exemplo

2

NUMBER_OF_EFFECTS

número de efeitos no modelo

Exemplo

5

OBSERVATION(S)

coluna(s) no arquivo de dados correspondentes às variáveis resposta

Detalhes

Deve-se informar tantas colunas quantas variáveis respostas houver, uma coluna por variável.

Exemplo

12 13 -> duas características, com valores nas colunas 12 e 13 do arquivo de dados.

10 -> uma característica, com valores na coluna 10 do arquivo de dados.

WEIGHT(S)

(coluna(s) onde estão as ponderações para as variáveis resposta)

Detalhes

Uma coluna por variável resposta; deixe em branco para análise sem ponderação.

Exemplo

```
10 11      > ponderações para a primeira e segunda característica estão
           presentes na coluna 10 e 11 do arquivo de dados,
           respectivamente.
```

```
-> deixe em branco para análise sem ponderação.
```

EFFECTS: POSITIONS_IN_DATAFILE NUMBER_OF_LEVELS TYPE_OF_EFFECT SAVE_
SAMPLES? [EFFECT NESTED]

posição(ões) no arquivo de dados - uma coluna por variável resposta, número de níveis, tipo de efeito, salvar amostras? **y/n**, (posição(ões) na qual o efeito está aninhado - uma coluna por variável resposta)

Detalhes

Seção dos efeitos no modelo com uma linha para cada efeito.

Os tipos de efeito possíveis no Intergen são:

- **cross** = classificatório,
- **cov** = covariável,
- **unknowncov** = efeito classificatório a ser usado como covariável para normas de reação via regressão aleatória,
- **rnorm** = normas de reação lineares via regressão aleatória,
- **ram** = efeito de modelo animal reduzido com reprodutores múltiplos,

A opção para salvar amostras **SAVE_SAMPLES?** y = sim, se aplica somente a análises MCMC e deve ser usada com cautela, especialmente para efeitos com muitos níveis, a fim de evitar a geração de arquivos muito grandes e, consequentemente, evitar que o algoritmo gaste muito tempo escrevendo em disco rígido.

A opção **[EFFECT NESTED]** é utilizada para especificar efeitos de regressão aleatória e de normas de reação com covariáveis desconhecidas, onde,

-> *posição no arquivo de dados* = a posição da covariável ou polinômio ortogonal (uma coluna por variável resposta);

-> *número de níveis* = número de níveis do efeito aninhado (p.ex.: efeito de animal),

-> *tipo de efeito* = **cov** para modelo de regressão aleatória e **rnorm** para modelo de normas de reação com cováriaveis desconhecidas;

-> *salvar amostras?* (y = sim, n = não);

-> *posição na qual o efeito está aninhado* = p. ex.: a posição do efeito animal no arquivo de dados (uma coluna por variável resposta).

Exemplo

4 1000 cov n 1 -> os valores do polinômio na coluna 4 do arquivo de dados, 1000 níveis para o efeito aninhado, tipo cov, não salvando amostras (que é a opção recomendada neste caso) e efeito aninhado (p. ex.: animal) na coluna 1.

1 1520 cross n

2 1520 cross n

3 1 cov n

4 50 cross n -> modelo univariado com quatro efeitos, três classificatórios e uma covariável.

Modelos diferentes para cada característica podem ser especificados. O valor zero (0) para a posição no arquivo de dados de um efeito é usado para codificar a ausência de tal efeito para uma característica em particular.

Exemplo (Misztal, 2012)

1 0 1000 cross n

0 7 2000 cross n -> Variável 1 tem grupo de contemporâneos (GC) na posição 1 com 1000 níveis e variável 2 tem GC na posição 7 com 2000 níveis.

1 7 2000 cross n -> Mesmo exemplo acima, em uma linha, onde o número de níveis é máximo entre os dois.

No caso de modelo animal reduzido (tipo de efeito = **ram**) esse(s) efeito(s) deve(m) ser o(s) primeiro(s) a aparecer(em) na lista da seção EFFECTS:, ou seja, deve estar na primeira linha da seção.

RANDOM_RESIDUAL: TYPE PRIOR_DEGREES_OF_BELIEF

tipo de distribuição residual, grau de confiança nos valores a priori

Detalhes

Os tipos de pressuposições possíveis para as distribuições dos resíduos são seis e são descritos em detalhe por Cardoso et al. (2005). Sucintamente, temos:

- **homogeneous** = Normal homoscedástico: Distribuição normal (gaussiana) com variância homogênea $e \sim N(0, \sigma^2_e)$ – que é a pressuposição usual;

IMPORTANTE: Essa é a única opção para análises multivariadas e análises BLUP. As seguintes são válidas somente para análises univariadas usando MCMC.

- **structural** = Normal heteroscedástico: Distribuição normal com variância heterogênea $e_{(i)} \sim N(0, \sigma^2_{e(i)})$, onde $\sigma^2_{e(i)}$ é variância específica dos erros $e_{(i)}$ de uma combinação de subclasses i definida por um modelo estrutural;

- **student_t** = Student t homoscedástico: Distribuição Student t com variância homogênea $e_{(i)} \sim N(0, \sigma^2_e/w_{(i)}) \rightarrow y_{(i)} \sim t_{(v)}(0, V^2_e)$ - modelo robusto;

- **struct_student_t** = Student t heteroscedástico: Distribuição Student t com variância heterogênea $e_{(i)} \sim N(0, \sigma^2_e/w_{(i)}) \rightarrow y_{(i)} \sim t_{(v)}(0, V^2_{e(i)})$ - modelo estrutural robusto;

- **slash** = Slash homoscedástico: Distribuição Slash com variância homogênea $e_{(i)} \sim N(0, \sigma^2_e/w_{(i)}) \rightarrow y_{(i)} \sim \text{Slash}_{(v)}(0, V^2_e)$ - modelo robusto;

- **struct_slash** = Slash heteroscedástico: Distribuição Slash com variância heterogênea $e_{(i)} \sim N(0, \sigma^2_e/w_{(i)}) \rightarrow y_{(i)} \sim \text{Slash}_{(v)}(0, V^2_{e(i)})$ - modelo estrutural robusto.

Para os graus de confiança nos valores a priori da(s) (co)variância(s) dos efeitos aleatórios, quanto maior o valor especificado, maior será a influência dos valores a priori nos resultados da análise. Especificando-se o número de variáveis na análise, é equivalente a uma priori difusa pouco informativa e, portanto, a inferência é basicamente a partir da informação dos dados.

Exemplo

homogeneous 1 $\rightarrow$ resíduos normais homocedásticos com 1 grau de confiança no valor a priori da variância residual.

METROPOLIS_STEP_OF_STRUCTURAL_EFFECTS: ROUNDS_WITHIN_CYCLE TUNING_SKIP
(opcional)

número de passos de Metropolis, intervalo de sintonia

Detalhes

Esta seção opcional é necessária somente se houver efeito estrutural nos resíduos, tipos *structural*, *struct_student_t* ou *struct_slash*.

Nela é especificado o número de passos do algoritmo de Metropolis-Hastings (MH) dentro de cada ciclo de MCMC e intervalo de ciclos para ajustar a variância da distribuição de propostas no algoritmo de MH durante o período de aquecimento (opções avançadas do modelo estrutural).

Exemplo

5 10

NUMBER_OF_STRUCTURAL_EFFECTS (opcional)

número de efeitos para a variância residual

Detalhes

Esta seção opcional é necessária somente se houver efeito estrutural nos resíduos, tipos *structural*, *struct_student_t* ou *struct_slash*.

Nela são definidos quantos efeitos do modelo também afetam a variância residual.

Exemplo

4

STRUCTURAL_EFFECTS: LINE_FROM_EFFECTS_SECTION SAVE_SAMPLES? (opcional)

linha da seção EFFECTS, salvar amostras? (**y/n**)

Detalhes

Esta seção opcional é necessária somente se houver efeito estrutural nos resíduos, tipos *structural*, *struct_student_t* ou *struct_slash*, devendo ter uma linha para cada efeito.

A linha da seção EFFECTS, se refere ao número de ordem da linha dessa seção de efeitos no modelo linear onde se especificou o efeito que vai afetar também a dispersão dos resíduos.

Pode-se salvar amostras desses fatores escalares usando **y** (= sim).

Exemplo

4 y -> usar efeito da quarta linha da seção EFFECTS: e salvar amostras.

6 y

7 y

9 n -> efeito da nona linha da seção EFFECTS: e não salvar amostras

RESIDUAL_PRIOR_(CO)VARIANCES

valores a priori para a matriz de covariância residual

Exemplo

200. 10.0

10.0 100. -> Para um modelo bivariado variâncias a priori na diagonal e covariâncias fora dela.

200 0.0

0.0 100

RANDOM_GROUP (opcional)

linha(s) do(s) efeito(s) aleatórios na seção EFFECTS:

Detalhes

Esta seção opcional é necessária quando existem efeitos aleatórios no modelo.

Pode-se especificar mais de uma linha, em caso de efeitos aleatórios correlacionados, por ex., genéticos aditivos direto e materno. Efeitos correlacionados devem necessariamente ser consecutivos na seção *EFFECTS*.

Outras seções descritas abaixo são necessárias para complementar a descrição dos efeitos aleatórios. A seção *RANDOM_GROUP* e as subsequentes são repetidas para cada efeito ou grupo de efeitos aleatórios.

Exemplo

1 2 -> efeitos na primeira e segunda linha da seção *EFFECTS* são aleatórios e correlacionados.

RANDOM_TYPE PRIOR_DEGREES_OF_BELIEF (opcional)

tipo aleatório, grau de confiança nos valores a priori

Detalhes

Esta seção opcional necessariamente acompanha a seção *RANDOM_GROUP*.

O tipo de efeito aleatório pode ser uma das seguintes possibilidades:

- **diagonal** = efeito não correlacionado,
- **add_sire** = efeito aditivo de touro,
- **add_animal** = efeito aditivo de animal,
- **add_an_ms** = efeito aditivo animal com reprodutores múltiplos e grupos genéticos,
- **add_an_mb** = efeito aditivo animal multirracial com grupos genéticos,
- **diag_mb** = efeito não correlacionado multirracial,
- **add_an_upg** = efeito aditivo com grupos genéticos,
- **user_file** = matriz de correlação definida pelo usuário (matriz não é invertida pelo programa),
- **user_file_i** = matriz de correlação definida pelo usuário (matriz é invertida pelo programa),
- **norm_prior** = efeito *normal prior* para os efeitos fixos.

Exemplo

add_animal 2 -> efeito de modelo animal com 2 graus de confiança no valor a priori.

PEDIGREEFILE: NAME N_ANIMAL N_GENETIC_GROUPS [N_BREEDS] (opcional)

(nome do arquivo de pedigree, número de animais), (número de grupos genéticos), (número de raças)

Detalhes

Esta seção opcional necessariamente acompanha a seção *RANDOM_GROUP*.

Os diferentes tipos de efeitos aleatórios requerem diferentes arquivos de pedigree. De forma geral, o arquivo de pedigree deve ter uma linha por animal com pelo menos três colunas.

Todos os animais devem constar no arquivo de pedigree identificados de 1 ao número total de animais no arquivo, incluindo os animais base sem pais conhecidos.

Não há a necessidade de os pais terem número menor que os filhos, exceto para o modelo animal reduzido (tipo de efeito = **ram** e tipo aleatório = **add_an_ms**).

Quando pai, mãe ou ambos forem desconhecidos devem ser identificados com 0. Outras colunas serão necessárias dependendo do tipo de efeito aleatório associado com o pedigree em questão, conforme descrito a seguir:

- **diagonal** ou **diag_mb**:

Não requer arquivo de pedigree, pois os níveis não são correlacionados dentro de variável. Ex: grupos contemporâneos aleatórios e efeito de ambiente permanente da vaca.

- **add_sire**:

número do animal, número do pai, número do avô materno

- **add_animal**:

número do animal, número do pai, número da mãe

- **add_an_ms**:

número do animal, número da mãe, número do pai (ou -1 se for animal com incerteza de paternidade - reprodutores múltiplos), indicador se o animal tem progênie no arquivo (1=sim, 0=não).

- **add_an_mb**:

número do animal, número do pai, número da mãe, proporção da raça 1,..., proporção da raça n.

Grupos genéticos (fixos) estimados conjuntamente com os efeitos aditivos dos animais nos tipos **add_an_ms** e **add_an_mb** são especificados atribuindo-se a eles números maiores que o número de animais no arquivo de pedigree. Por exemplo, se há 100 animais no total e desejamos criar dois grupos genéticos, os animais são associados a eles através de pais fictícios com números 101 e 102, dependendo do grupo ao qual pertencem.

Exemplo

`mbreed1.ped 100 2 2` -> no arquivo *mbreed1.ped* existem 100 animais com dois grupos genéticos de uma população formada por duas raças.

Uma possível linha desse arquivo seria:

`23 101 101 0.625 0.375` -> para um animal número 23, com pais desconhecidos, mas associado ao grupo genético 101, é 5/8 da raça 1 e 3/8 da raça 2.

MULTIPLE_SIRES: MAX_N_FOR_MCMC [FILE: NAME & DIMENSION] [DIRICHLET_PRIORS] (opcional)

número máximo de reprodutores múltiplos (RM) para inferência MCMC, (nome do arquivo de RM, dimensão do arquivo), (hiperparâmetros da distribuição a priori Dirichlet)

Detalhes

Seção opcional, acompanha a seção *RANDOM_GROUP* somente se o tipo de efeito aleatório = **add_an_ms** ou **add_an_mb**, que define inferência na incerteza de paternidade quando há RM. O número máximo de possíveis pais para fazer inferência define o tipo de inferência com respeito às probabilidades de paternidade. Deve-se usar 0 para análise sem RM com os tipos aleatórios **add_an_ms** e **add_an_mb**, e usar 1 para utilizar a matriz de parentesco médio, que também será usada quando o tamanho do grupo de RM exceder o número máximo de possíveis pais para fazer inferência MCMC.

Se a análise incluir animais com RM deve-se informar o *nome e o número de linhas do arquivo de RM*. Nesse arquivo são indicados quais os possíveis pais para cada indivíduo associado a um grupo RM por ter o valor -1 na coluna do número do pai no arquivo de pedigree. O arquivo de RM tem a seguinte estrutura para cada animal:

número de possíveis pais

identificação do pai 1

probabilidade a priori do pai 1

...

identificação do pai n

probabilidade a priori do pai n

Por exemplo, para um animal com 3 pais possíveis (4, 5 ou 6) com iguais probabilidades de paternidade a priori, temos,

3

4

0.3333333333

5

0.3333333333

6

0.3333333333

O programa INTERGEN, cada vez que encontra um animal com incerteza de paternidade no arquivo de pedigree (identificado pelo valor - 1 na coluna do pai), busca no arquivo de RM quais os possíveis pais. Portanto, os possíveis pais de cada animal devem aparecer no *arquivo de RM* na mesma ordem que esses animais com paternidade incerta são relacionados no *arquivo de pedigree*. A estrutura descrita acima deve ser repetida para cada um desses animais e a identificação dos pais deve ser consistente com a do arquivo de pedigree.

Finalmente, nesta seção, o valor para hiperparâmetros da distribuição a priori Dirichlet das probabilidades de paternidade define as priores para as probabilidades de paternidade. As opções determinadas conforme esse valor são:

- se = 0, não usa a distribuição Dirichlet;
- se > 0, usa o valor especificado para todos os parâmetros alfa da distribuição Dirichlet em todos os grupos de RM, e
- se < 0, usa valores definidos pelo usuário no arquivo de RM para especificar os parâmetros alfas.

Exemplo

10 animodel.ms 252 -1 -> Faz dois ciclos de MH dentro de cada ciclo da cadeia MCMC e ajusta a variância da distribuição de propostas baseada na taxa de aceitação média a cada 20 ciclos durante a primeira metade do período de aquecimento.

- add_an_upg:

número do animal, número do pai, número da mãe, 3 menos o número de pais conhecidos (3 = ambos os pais desconhecidos, 2 = pelo menos um dos pais conhecidos, 1 = ambos os pais conhecidos).

Os grupos genéticos são identificados com números maiores que o número de animais no pedigree. Por exemplo, se há 100 animais no pedigree e desejamos criar dois grupos genéticos, os animais são associados a eles através de pais fictícios com números 101 e 102, dependendo do grupo ao qual pertencem.

Exemplo:

geneticgroups.ped 100

Uma possível linha desse arquivo seria:

23 101 101 3 -> para um animal número 23, com pais desconhecidos, mas associado ao grupo genético 101 e possui ambos os pais desconhecidos.

- user_file ou user_file_i:

Quando uma matriz de relacionamento não pode ser criada pelo programa, existe a possibilidade de o usuário criá-la separadamente e incluí-la. A opção **user_file** incorpora a matriz diretamente (não inverte a matriz), enquanto a opção **user_file_i** incorpora a inversa da matriz fornecida (importa a matriz e a inverte internamente).

Exemplo

dominance.ped

O exemplo abaixo representa uma matriz simétrica qualquer 6 x 6 para um efeito aleatório com 6 níveis:

0.50	0.30	0	1.50	2.30	5.55
0.30	2.00	6.30	0.21	2.15	4.39
0.35	6.30	1.15	0.68	0.75	0.99
1.50	0.21	0.68	1.75	2.34	4.55
5.55	2.15	0.75	2.34	3.10	2.22
5.55	4.39	0.99	4.55	2.22	0.80

O programa utiliza as informações contidas na matriz acima, porém o arquivo fornecido pelo usuário deve necessariamente apresentar a estrutura abaixo. Nesse formato somente são fornecidos os valores localizados no triângulo superior (ou inferior) da matriz com a respectiva linha (primeira coluna do arquivo) e coluna (segunda coluna do arquivo) do valor (terceira coluna).

1	1	0.50
1	2	0.30
1	4	1.50
1	5	2.30
1	6	5.55
2	2	2.00
2	3	6.30
2	4	0.21
⋮	⋮	⋮
5	6	2.22
6	6	0.80

Observação: note que o valor localizado na primeira linha e terceira coluna da matriz é igual a zero e não foi incluído no arquivo fornecido. Valores iguais a zero não precisam ser incluídos.

- **norm_prior:**

característica, nível do efeito fixo, valor a priori do respectivo nível do efeito fixo na característica, variância do valor a priori para o respectivo nível do efeito fixo.

Esse efeito permite incluir uma informação a priori sobre o valor esperado (μ_p e a variância (V_p para cada nível dos efeitos fixos do modelo. O valor μ_p pode ser interpretado como a “melhor” estimativa prévia antes de observar os dados a serem analisados e V_p como a certeza sobre a estimativa. Assim, menores valores de V_p são sugestivos de maior confiança do valor *a priori*, enquanto maiores valores sugerem incerteza. Dessa forma, o sistema de equações tradicional, $Cb=y$, é ajustado para $(C+V_p^{-1})b=y+V_p^{-1} \mu_p$ quando uma informação *a priori* é incorporada. O efeito pode

ser ajustado em modelos uni e multi-características e utiliza um arquivo externo para ler as informações (*a priori*) de média e variância para **todos** os níveis do efeito fixo da(s) característica(s).

Exemplo: Considere um modelo bi-característica em que determinado efeito fixo tenha três níveis para a primeira característica e dois níveis para a segunda característica. O arquivo fornecido pelo usuário **deve** necessariamente ter a seguinte estrutura.

1	1	20.0	5.00
1	2	14.0	20.0
1	3	5.30	2.00
2	1	5.60	2.20
2	2	3.10	22.7
2	3	0.00	0.00

A primeira coluna do arquivo relaciona a característica, a terceira relaciona os níveis do efeito fixo dentro da característica, a quarta é a média *a priori* e a quarta coluna representa a variância.

Observação: note que, pelo exemplo, a segunda característica possui apenas dois níveis, porém foi fornecida uma linha para o terceiro nível da segunda característica com valores de média e variância iguais a zero.

(CO)VARIANCES (opcional)

valores *a priori* para a matriz de covariância dos efeitos aleatórios

Detalhes

Seção opcional que necessariamente acompanha a seção *RANDOM_GROUP*. A exceção é quando for especificado *RANDOM_TYPE* igual a *norm_prior*.

A estrutura da matriz de (co)variância (*G*) permite múltiplos efeitos aleatórios e múltiplas características. Considerando um exemplo com dois efeitos (*a* e *m*) e duas características (1 e 2), temos a seguinte matriz simétrica, onde se tem blocos diagonais de cada efeito com variâncias e covariâncias entres as diferentes características e blocos fora da diagonal com as covariâncias entre os diferentes efeitos e/ou características:

$$G = \begin{bmatrix} \sigma_{a_1}^2 & \sigma_{a_1a_2} & \sigma_{a_1m_1} & \sigma_{a_1m_2} \\ \sigma_{a_1a_2} & \sigma_{a_2}^2 & \sigma_{a_2m_1} & \sigma_{a_2m_2} \\ \sigma_{a_1m_1} & \sigma_{a_2m_1} & \sigma_{m_1}^2 & \sigma_{m_1m_2} \\ \sigma_{a_1m_2} & \sigma_{a_2m_2} & \sigma_{m_1m_2} & \sigma_{m_2}^2 \end{bmatrix}$$

Embora essas matrizes sejam simétricas, todos seus os elementos devem ser especificados na seção (CO)VARIANCES.

Exemplo

200 10.00 10.0 0.000

100 100.0 0.00 15.00

100 0.000 50.0 2.000

0.0 15.00 2.00 300.0 -> Note que $\sigma_{a_1m_2} = \sigma_{a_2m_1} = 0$, isto é, a covariância é assumida nula entre os efeitos *a* e *m* nas diferentes características e mantida em zero (0) durante a análise.

No caso de modelo multirracial (RANDOM_TYPE = **add_an_mb** ou **diag_mb**), a seção (CO) VARIANCES deve conter as variâncias específicas por raça e a variância da segregação entre raças. A matriz é especificada em grandes blocos para raças na diagonal e para segregação fora dela, conforme o exemplo abaixo, no qual os mesmos efeitos (*a* e *m*) e características (1 e 2) acima, são especificados para duas raças (*b* e *c*):

No caso de modelo multirracial (RANDOM_TYPE = **add_an_mb** ou **diag_mb**), a seção (CO) VARIANCES deve conter as variâncias específicas por raça e a variância da segregação entre raças. A matriz é especificada em grandes blocos para raças na diagonal e para segregação fora dela, conforme o exemplo abaixo, no qual os mesmos efeitos (*a* e *m*) e características (1 e 2) acima, são especificados para duas raças (*b* e *c*):

$$G = \begin{bmatrix} \sigma_{a_1b}^2 & \sigma_{a_1a_2b} & \sigma_{a_1m_1b} & \sigma_{a_1m_2b} & \sigma_{a_1S_{bc}}^2 & \sigma_{a_1a_2S_{bc}} & \sigma_{a_1m_1S_{bc}} & \sigma_{a_1m_2S_{bc}} \\ \sigma_{a_1a_2b} & \sigma_{a_2b}^2 & \sigma_{a_2m_1b} & \sigma_{a_2m_2b} & \sigma_{a_1a_2S_{bc}} & \sigma_{a_2S_{bc}}^2 & \sigma_{a_2m_1S_{bc}} & \sigma_{a_2m_2S_{bc}} \\ \sigma_{a_1m_1b} & \sigma_{a_2m_1b} & \sigma_{m_1b}^2 & \sigma_{m_1m_2b} & \sigma_{a_1m_1S_{bc}} & \sigma_{a_2m_1S_{bc}} & \sigma_{m_1S_{bc}}^2 & \sigma_{m_1m_2S_{bc}} \\ \sigma_{a_1m_2b} & \sigma_{a_2m_2b} & \sigma_{m_1m_2b} & \sigma_{m_2b}^2 & \sigma_{a_1m_2S_{bc}} & \sigma_{a_2m_2S_{bc}} & \sigma_{m_1m_2S_{bc}} & \sigma_{m_2b}^2 \\ \sigma_{a_1S_{bc}}^2 & \sigma_{a_1a_2S_{bc}} & \sigma_{a_1m_1S_{bc}} & \sigma_{a_1m_2S_{bc}} & \sigma_{a_1c}^2 & \sigma_{a_1a_2c} & \sigma_{a_1m_1c} & \sigma_{a_1m_2c} \\ \sigma_{a_1a_2S_{bc}} & \sigma_{a_2S_{bc}}^2 & \sigma_{a_2m_1S_{bc}} & \sigma_{a_2m_2S_{bc}} & \sigma_{a_1a_2c} & \sigma_{a_2c}^2 & \sigma_{a_2m_1c} & \sigma_{a_2m_2c} \\ \sigma_{a_1m_1S_{bc}} & \sigma_{a_2m_1S_{bc}} & \sigma_{m_1S_{bc}}^2 & \sigma_{m_1m_2S_{bc}} & \sigma_{a_1m_1c} & \sigma_{a_2m_1c} & \sigma_{m_1c}^2 & \sigma_{m_1m_2c} \\ \sigma_{a_1m_2S_{bc}} & \sigma_{a_2m_2S_{bc}} & \sigma_{m_1m_2S_{bc}} & \sigma_{m_2b}^2 & \sigma_{a_1m_2c} & \sigma_{a_2m_2c} & \sigma_{m_1m_2c} & \sigma_{m_2c}^2 \end{bmatrix}$$

Note que a matriz **G** acima é uma maneira de especificar para o programa todos os componentes de variância das raças **b** e **c** e da segregação entre essas raças, **S_{bc}**, mas não é a matriz de covariância multirracial, a qual será montada internamente pelo programa considerando também as informações de parentesco.

Como rodar o programa

Para rodar o programa, primeiro deve-se colocar os arquivos de parâmetros, de dados e de pedigree (e outros arquivos dependendo do tipo de efeitos definidos) e o executável (*intergen1.3.exe* para Windows ou *intergen1.3* para Linux/Mac OS) no mesmo diretório. Depois abrir uma janela de command prompt do DOS ou terminal do Linux/Mac OS, ir até o diretório onde foram colocados os arquivos e digitar: *intergen1.3* (no Windows) ou *./intergen1.3* (no Linux/Mac OS) <enter>. O programa então irá solicitar o nome do arquivo de parâmetros, através da seguinte pergunta:

name of parameter file?

Deve-se então digitar o nome do arquivo de parâmetros <enter>.

Essa resposta pode ser incluída em um arquivo texto (digamos *run*), p. ex.: o arquivo *run* (ASCII) tem o seguinte conteúdo:

parameterfile

Assim podemos chamar o programa da seguinte forma:

```
C:\...caminho...\intergen1.3 < run > out <enter>
```

Neste caso, o INTERGEN obtém o nome do arquivo de parâmetros do arquivo *run* e toda a saída de tela é colocada no arquivo *out*.

IMPORTANTE: Muitas paradas anormais do programa podem estar associadas a erros no número de níveis dos efeitos do modelo e na especificação das posições dos efeitos. Muitas vezes esses erros vêm associados com a palavra “hash” ou “ACCESS VIOLATION”.

Como continuar uma cadeia já iniciada

Em inferência MCMC, é comum necessitar alongar uma cadeia já finalizada para aumentar o conteúdo de informação ou por ter uma cadeia interrompida por falta de energia ou pela necessidade de desligar o computador. Nestes casos, o INTERGEN através da seção `RESTART: Y/N? [CYCLE_TO_RESTART]` oferece a oportunidade de continuar a cadeia por mais ciclos, sem ter que passar por novo período de aquecimento ou perder os ciclos de uma cadeia interrompida.

Para tal, o programa, cada vez que grava os arquivos de saída, grava também a última amostra de todos os parâmetros nos arquivos *solutions*, *varcomp*, *structural_r* e *robustness_w*.

Para continuar/reiniciar a cadeia, o usuário deve observar na primeira linha de um dos arquivos acima qual foi o último ciclo gravado, por exemplo, no *varcomp*:

```
Variance components after: 44000 rounds and burn-in of 10000.
```

Neste caso o último ciclo foi o 44.000 e, portanto, a cadeia deve ser reiniciada no 44.001, alterando a seção correspondente no arquivo de parâmetros da seguinte forma:

```
RESTART: Y/N? [CYCLE_TO_RESTART]
```

```
y 44001
```

IMPORTANTE: Existe um passo não automático muito importante de ser completado pelo usuário. Nos arquivos *solutionsam*, *varcompsam*, e *loglike_rnd* as amostras são salvas com maior frequência que nos outros acima e, portanto, o usuário deverá abrir manualmente com um editor de textos esses arquivos, ir até o final do arquivo e apagar todas as amostras salvas após o último ciclo gravado nos arquivos *solutions*, *varcomp*, *structural_r* e *robustness_w* (44.000 no exemplo acima), salvando os arquivos com os nomes originais sem colocar nenhuma extensão.

Arquivos de saída do programa

Os seguintes arquivos de saída são gerados pelo programa INTERGEN após executar análises MCMC ou BLUP:

Solutions (MCMC): Média e desvio padrão a posteriori dos efeitos no modelo após o período de aquecimento e última amostra para continuar cadeia.

Solutions (BLUP): Solução e desvio padrão do erro de predição dos efeitos no modelo.

Solutionsam (MCMC): Amostras dos efeitos no modelo na seção EFFECTS onde opção SAVE_SAMPLES = y.

Varcomp (MCMC): Média e desvio padrão a posteriori dos componentes de variância após o período de aquecimento e última amostra para recomençar cadeia. Os efeitos dos RANDOM_GROUP aparecem primeiro e na mesma ordem que são especificados no arquivo parameterfile e por últimos aparecem os componentes de (co)variância residual.

Varcompsam (MCMC): Amostras dos componentes de variância e, se for o caso, efeitos do modelo estrutural heteroscedástico. Os componentes de variância aparecem na mesma ordem do varcomp e dessas matrizes, por serem simétricas, são listados somente elementos da diagonal inferior, da seguinte forma: v11, v21, v22, v13, v23, v33... Os fatores multiplicativos dos efeitos estruturais para variância residual aparecem após a variância residual e na mesma ordem que são especificados no parameterfile.

structural_r (MCMC): Média e desvio padrão a posteriori dos efeitos estruturais na variância residual após o período de aquecimento e última amostra para recomençar cadeia, listados na mesma ordem que são especificados no parameterfile.

robustness_w (MCMC): Média e desvio padrão a posteriori das variáveis de ponderação do modelo robusto após o período de aquecimento e última amostra para recomençar cadeia, para cada registro, na mesma ordem do arquivo de dados.

Loglike_rnd (MCMC): O logaritmo da distribuição condicional de geração dos dados (primeiro estágio do modelo hierárquico) para critérios de escolha de modelo e medidas de ajuste.

Loglike_obs (MCMC): Informações para critérios de escolha de modelo e medidas de ajuste.

Mh_dbeliefchg (MCMC): Sintonia do fator escalar multiplicativo da variância das distribuições proposta do algoritmo de MH (se for o caso).

Mh_acceptance (MCMC): Taxa de aceitação de valores propostos no algoritmo de MH (se for o caso).

Msiresppr (MCMC): Média e desvio padrão a posteriori das probabilidades de paternidade após o período de aquecimento e última amostra para recomençar cadeia (se for o caso). Este arquivo ainda salva a proporção de vezes que cada pai candidato foi escolhido com pai verdadeiro no ciclo MCMC.

Msirespprsam (MCMC): Amostras das probabilidades de paternidade para os possíveis pais do primeiro animal que aparece com incerteza de paternidade no arquivo de pedigree.

yhat_residual (BLUP): Estimativas do fenótipo predito ($\hat{y}=y-\hat{e}$), fenótipo ajustado ($\hat{y}=y-X\hat{b}$) e resíduo ($\hat{e}$).

dinv_elements: diagonal da matriz dos numeradores de parentesco de Wright (u), inversa da raiz quadrada da diagonal da matriz de variância de segregação mendeliana (v) e soma dos produtos entre as probabilidades de pai e mãe (w) de cada indivíduo do arquivo de pedigree.

Referências

CARDOSO, F. F. **Aplicação da inferência bayesiana no melhoramento animal usando o programa Intergen**: manual da versão 1.2. Bagé: Embrapa Pecuária Sul, 2010. 31 p. (Embrapa Pecuária Sul. Documentos, 111).

CARDOSO, F. F. **Manual de utilização do programa Interger - versão 1.0 em estudos de genética quantitativa animal**. Bagé: Embrapa Pecuária Sul, 2008. 74 p. (Embrapa Pecuária Sul. Documentos, 74).

CARDOSO, F. F.; ROSA, G. J. M.; TEMPELMAN, R. J. Multiple breed genetic inference using heavy-tailed structural models for heterogeneous residual variances. **Journal of Animal Science**, Savoy, v. 83, n. 8, p. 1766-1779, Aug. 2005.

HENDERSON, C. R. **Applications of linear models in animal breeding**. Guelph: University of Guelph. 1984. 462 p.

MISZTAL, I.; TSURUTA, S.; STRABEL, T.; AUVRAY, B.; DRUET, T.; LEE, D. H. BLUPF90 and related programs (BGF90). In: WORLD CONGRESS ON GENETICS APPLIED TO LIVESTOCK PRODUCTION, 7., 2002, Montpellier. **Proceedings...** Montpellier: INRA: CIRAD, 2002. 1 CD-ROM. (Session 28. Communication n. 28-07).

MISZTAL, I. **BLUPF90 - a flexible mixed model program in Fortran 90**. [Athens]: University of Georgia, 2012. Disponível em: <<http://nce.ads.uga.edu/wiki/lib/exe/fetch.php?media=blupf90.pdf>>. Acesso em: 18 fev. 2017.

SORENSEN, D. A.; GIANOLA, D. **Likelihood, Bayesian and MCMC methods in quantitative genetics**. New York: Springer-Verlag, 2002. 740 p.

Anexos

A seguir é apresentada uma série de arquivos de parâmetros para rodar determinados modelos, frequentemente utilizados em análises de genética quantitativa baseada em métodos bayesianos e BLUP. Os exemplos incluindo arquivos de dados, pedigree e parâmetros, estão disponíveis na página da Embrapa Pecuária Sul .

A.1. Modelo animal com prioris próprias

```
# Exemplo 1 - Modelo animal

MCMC_CHAIN: TOTAL_CYCLES BURN_IN THINNING_INTERVAL

11000 1000 10

SEED

2

RESTART: Y/N? [CYCLE_TO_RESTART]

n

DATAFILE NAME N_RECORDS

animodel.dat 240

NUMBER_OF_TRAITS

1

NUMBER_OF_EFFECTS

2

OBSERVATION(S)

4
```

WEIGHT(S)

EFFECTS: POSITIONS_IN_DATAFILE NUMBER_OF_LEVELS TYPE_OF_EFFECT SAVE_
SAMPLES? [EFFECT NESTED]

1 300 cross n #efeito_do_animal

3 2 cross n #sexo

RANDOM_RESIDUAL: TYPE PRIOR_DEGREES_OF_BELIEF

homogeneous 1

RESIDUAL_PRIOR_(CO)VARIANCES

150

RANDOM_GROUP

1

RANDOM_TYPE PRIOR_DEGREES_OF_BELIEF

add_animal 1

PEDIGREEFILE: NAME N_ANIMAL N_GENETIC_GROUPS [N_BREEDS]

animodel.ped 300 0

(CO)VARIANCES

100

A.2. Modelo animal utilizando a inversa da matriz de parentesco genômico

Exemplo 2 - Modelo animal utilizando seleção genômica

MCMC_CHAIN: TOTAL_CYCLES BURN_IN THINNING_INTERVAL

11000 1000 10

SEED

2

RESTART: Y/N? [CYCLE_TO_RESTART]

n

DATAFILE NAME N_RECORDS

animodel.dat 240

NUMBER_OF_TRAITS

1

NUMBER_OF_EFFECTS

2

OBSERVATION(S)

4

WEIGHT(S)

EFFECTS: POSITIONS_IN_DATAFILE NUMBER_OF_LEVELS TYPE_OF_EFFECT SAVE_SAMPLES? [EFFECT NESTED]

1 300 cross n #efeito_do_animal

3 2 cross n #sexo

RANDOM_RESIDUAL: TYPE PRIOR_DEGREES_OF_BELIEF

homogeneous 1

RESIDUAL_PRIOR_(CO)VARIANCES

150

RANDOM_GROUP

1

RANDOM_TYPE PRIOR_DEGREES_OF_BELIEF

user_file 1

PEDIGREEFILE: NAME N_ANIMAL N_GENETIC_GROUPS [N_BREEDS]

Hinv.txt

(CO)VARIANCES

100

A.3. Modelo animal usando uma matriz do numerador do parentesco médio de Henderson.

Exemplo 3 - Modelo animal com matriz de parentesco medio

MCMC_CHAIN: TOTAL_CYCLES BURN_IN THINNING_INTERVAL

11000 1000 10

SEED

2

RESTART: Y/N? [CYCLE_TO_RESTART]

n

DATAFILE NAME N_RECORDS

animodel.dat 240

NUMBER_OF_TRAITS

1

NUMBER_OF_EFFECTS

2

OBSERVATION(S)

4

WEIGHT(S)

EFFECTS: POSITIONS_IN_DATAFILE NUMBER_OF_LEVELS TYPE_OF_EFFECT SAVE_
SAMPLES? [EFFECT NESTED]

1 300 cross n #efeito_do_animal

3 2 cross n #sexo

RANDOM_RESIDUAL: TYPE PRIOR_DEGREES_OF_BELIEF

homogeneous 10000

RESIDUAL_PRIOR_(CO)VARIANCES

150

RANDOM_GROUP

1

RANDOM_TYPE PRIOR_DEGREES_OF_BELIEF

add_an_ms 1

PEDIGREEFILE: NAME N_ANIMAL N_GENETIC_GROUPS [N_BREEDS]

animodelrm.peda 300 0

MULTIPLE_SIRES: MAX_N_FOR_MCMC [FILE: NAME & DIMENSION] [DIRICHLET_PRIORS]

1 animodelrm.ms 252

(CO)VARIANCES

100

A.4. Modelo animal com grupos genéticos de Westell.

```

# Exemplo 4 - Modelo animal com Grupos Geneticos

MCMC_CHAIN: TOTAL_CYCLES BURN_IN THINNING_INTERVAL

11000 1000 10

SEED

2

RESTART: Y/N? [CYCLE_TO_RESTART]

n

DATAFILE NAME N_RECORDS

animodel.dat 240

NUMBER_OF_TRAITS

1

NUMBER_OF_EFFECTS

2

OBSERVATION(S)

4

WEIGHT(S)

EFFECTS: POSITIONS_IN_DATAFILE NUMBER_OF_LEVELS TYPE_OF_EFFECT SAVE_
SAMPLES? [EFFECT NESTED]

1 301 cross n #efeito_do_animal

3 2 cross n #sexo

RANDOM_RESIDUAL: TYPE PRIOR_DEGREES_OF_BELIEF

homogeneous 10000

RESIDUAL_PRIOR_(CO)VARIANCES

150

RANDOM_GROUP

1

RANDOM_TYPE PRIOR_DEGREES_OF_BELIEF

add_an_ms 1

PEDIGREEFILE: NAME N_ANIMAL N_GENETIC_GROUPS [N_BREEDS]

```

```

animodelrm.pedg 300 1

MULTIPLE_SIRES: MAX_N_FOR_MCMC [FILE: NAME & DIMENSION] [DIRICHLET_PRIORS]

0

(CO)VARIANCES

100

```

A.5. Modelo de incerteza de paternidade para filhos de acasalamentos com reprodutores múltiplos

```

# Exemplo 5 - Modelo de incerteza de paternidade

MCMC_CHAIN: TOTAL_CYCLES BURN_IN THINNING_INTERVAL

11000 1000 10

SEED

2

RESTART: Y/N? [CYCLE_TO_RESTART]

n

DATAFILE NAME N_RECORDS

animodel.dat 240

NUMBER_OF_TRAITS

1

NUMBER_OF_EFFECTS

2

OBSERVATION(S)

4

WEIGHT(S)

EFFECTS: POSITIONS_IN_DATAFILE NUMBER_OF_LEVELS TYPE_OF_EFFECT SAVE_
SAMPLES? [EFFECT NESTED]

1 300 cross n #efeito_do_animal

3 2 cross n #sexo

RANDOM_RESIDUAL: TYPE PRIOR_DEGREES_OF_BELIEF

homogeneous 10000

```



```

RESIDUAL_PRIOR_ (CO) VARIANCES
150
RANDOM_GROUP
1
RANDOM_TYPE PRIOR_DEGREES_OF_BELIEF
add_an_ms 1
PEDIGREEFILE: NAME N_ANIMAL N_GENETIC_GROUPS [N_BREEDS]
animodelrm.ped 300 0
MULTIPLE_SIRES: MAX_N_FOR_MCMC [FILE: NAME & DIMENSION] [DIRICHLET_PRIORS]
10 animodelrm.ms 252 -1
(CO) VARIANCES
100

```

A.6. Modelo estrutural para variâncias residuais, com heterogeneidade de variância e robustez a dados extremos

```

# Exemplo 6 - Modelo animal heteroscedastico robusto
MCMC_CHAIN: TOTAL_CYCLES BURN_IN THINNING_INTERVAL
11000 1000 10
SEED
2
RESTART: Y/N? [CYCLE_TO_RESTART]
n
DATAFILE NAME N_RECORDS
animodelh.dat 240
NUMBER_OF_TRAITS
1
NUMBER_OF_EFFECTS
2
OBSERVATION(S)
4
WEIGHT(S)

```

```

EFFECTS: POSITIONS_IN_DATAFILE NUMBER_OF_LEVELS TYPE_OF_EFFECT SAVE_
SAMPLES? [EFFECT NESTED]

1 300 cross n #efeito_do_animal

3 2 cross n #sexo

RANDOM_RESIDUAL: TYPE PRIOR_DEGREES_OF_BELIEF

struct_student_t 1

METROPOLIS_STEP_OF_STRUCTURAL_EFFECTS: ROUNDS_WITHIN_CYCLE TUNING_SKIP

1 10

NUMBER_OF_STRUCTURAL_EFFECTS

1

STRUCTURAL_EFFECTS: LINE_FROM_EFFECTS_SECTION SAVE_SAMPLES?

2 y

RESIDUAL_PRIOR_(CO)VARIANCES

150

RANDOM_GROUP

1

RANDOM_TYPE PRIOR_DEGREES_OF_BELIEF

add_animal 1

PEDIGREEFILE: NAME N_ANIMAL N_GENETIC_GROUPS [N_BREEDS]

animodelh.ped 300 0

(CO)VARIANCES

100

```

A.7. Modelo animal multirracial

```

# Exemplo 7 - Modelo multirracial

MCMC_CHAIN: TOTAL_CYCLES BURN_IN THINNING_INTERVAL

4000 2000 20

SEED

123

RESTART: Y/N? [CYCLE TO RESTART]

```

n

DATAFILE NAME N_RECORDS

mbreed1.dat 4000

NUMBER_OF_TRAITS

1

NUMBER_OF_EFFECTS

3

OBSERVATION(S)

4

WEIGHT(S)

EFFECTS: POSITIONS_IN_DATAFILE NUMBER_OF_LEVELS TYPE_OF_EFFECT [EFFECT
NESTED]

1 4000 cross n

2 1 cov n

3 1 cov n

RANDOM_RESIDUAL: TYPE PRIOR_DEGREES_OF_BELIEF

homogeneous 1

RESIDUAL_PRIOR_(CO)VARIANCES

100

RANDOM_GROUP

1

RANDOM_TYPE PRIOR_DEGREES_OF_BELIEF

add_an_mb 3

PEDIGREEFILE: NAME N_ANIMAL N_GENETIC_GROUPS [N_BREEDS]

mbreed1.ped 4000 0 2

MULTIPLE_SIRES: MAXN_FULL [FILE: NAME DIMENSION] [ALPHA_PRIORS]

0

METROPOLIS_STEP_OF_MULTIBREED_(CO)VARIANCES: ROUNDS_WITHIN_CYCLE
TUNING_SKIP

5 10

(CO)VARIANCES

100 20

20 50

A.8. Modelo de normas de reação

Exemplo 8 - Modelo de Normas de Reação

MCMC_CHAIN: TOTAL_CYCLES BURN_IN THINNING_INTERVAL

20000 5000 30

SEED

123

RESTART: Y/N? [CYCLE_TO_RESTART]

n

DATAFILE NAME N_RECORDS

rnorm_ex1.dat 1000

NUMBER_OF_TRAITS

1

NUMBER_OF_EFFECTS

4

OBSERVATION(S)

4

WEIGHT(S)

EFFECTS: POSITIONS_IN_DATAFILE NUMBER_OF_LEVELS TYPE_OF_EFFECT SAVE_
SAMPLES [EFFECT NESTED]

2 1 cov n

3 50 unknowncov n

2 1520 cov n 1

3 1520 rnorm n 1

RANDOM_RESIDUAL: TYPE PRIOR_DEGREES_OF_BELIEF

homogeneous 1

RESIDUAL_PRIOR_(CO)VARIANCES

```

100

RANDOM_GROUP

2

RANDOM_TYPE PRIOR_DEGREES_OF_BELIEF

diagonal 1

PEDIGREEFILE: NAME N_ANIMAL N_GENETIC_GROUPS [N_BREEDS]

(CO)VARIANCES

50

RANDOM_GROUP

3 4

RANDOM_TYPE PRIOR_DEGREES_OF_BELIEF

add_animal 2

PEDIGREEFILE: NAME N_ANIMAL N_GENETIC_GROUPS [N_BREEDS]

rnorm_ex1.ped 1520 0

(CO)VARIANCES

20 6

6 2

```

A.9. Modelo de normas de reação utilizando a inversa da matriz de parentesco genômico

```

# Exemplo 9 - Modelo de Normas de Reação utilizando seleção genômica

MCMC_CHAIN: TOTAL_CYCLES BURN_IN THINNING_INTERVAL

20000 5000 30

SEED

123

RESTART: Y/N? [CYCLE_TO_RESTART]

n

DATAFILE NAME N_RECORDS

rnorm_ex1.dat 1000

NUMBER_OF_TRAITS

1

```

NUMBER_OF_EFFECTS

4

OBSERVATION(S)

4

WEIGHT(S)

EFFECTS: POSITIONS_IN_DATAFILE NUMBER_OF_LEVELS TYPE_OF_EFFECT SAVE_
SAMPLES [EFFECT NESTED]

2 1 cov n

3 50 unknowncov n

2 1520 cov n 1

3 1520 rnorm n 1

RANDOM_RESIDUAL: TYPE PRIOR_DEGREES_OF_BELIEF

homogeneous 1

RESIDUAL_PRIOR_(CO)VARIANCES

100

RANDOM_GROUP

2

RANDOM_TYPE PRIOR_DEGREES_OF_BELIEF

diagonal 1

PEDIGREEFILE: NAME N_ANIMAL N_GENETIC_GROUPS [N_BREEDS]

(CO)VARIANCES

50

RANDOM_GROUP

3 4

RANDOM_TYPE PRIOR_DEGREES_OF_BELIEF

user_file 2

PEDIGREEFILE: NAME N_ANIMAL N_GENETIC_GROUPS [N_BREEDS]

Hinv.txt

(CO)VARIANCES

20 6

6 2

A.10. Modelo animal via BLUP

```
# Exemplo 10 - Modelo animal via BLUP

BLUP: SOLVE_METHOD CONVERGENCE_CRITERIA [STDERR]

PCG 1e-12 se

DATAFILE NAME N_RECORDS

animodel.dat 240

NUMBER_OF_TRAITS

1

NUMBER_OF_EFFECTS

2

OBSERVATION(S)

4

WEIGHT(S)


EFFECTS: POSITIONS_IN_DATAFILE NUMBER_OF_LEVELS TYPE_OF_EFFECT SAVE_
SAMPLES? [EFFECT NESTED]

1 300 cross n          # efeito_do_animal

3  2 cross n          #sexo

RANDOM_RESIDUAL: TYPE PRIOR_DEGREES_OF_BELIEF

homogeneous 1

RESIDUAL_PRIOR_(CO)VARIANCES

150

RANDOM_GROUP

1

RANDOM_TYPE PRIOR_DEGREES_OF_BELIEF

add_animal 1

PEDIGREEFILE: NAME N_ANIMAL N_GENETIC_GROUPS [N_BREEDS]

animodel.ped 300 0

(CO)VARIANCES
```

100

A.11. Modelo animal via BLUP com grupos genéticos de Westell sem modelagem de reprodutores múltiplos.

```
# Exemplo 11 - Modelo animal via BLUP com grupos genéticos de Westell
#               sem modelagem de reprodutores múltiplos
BLUP: SOLVE_METHOD CONVERGENCE_CRITERIA [STDERR]
PCG 1e-12 se
DATAFILE NAME N_RECORDS
animodel.dat 240
NUMBER_OF_TRAITS
1
NUMBER_OF_EFFECTS
2
OBSERVATION(S)
4
WEIGHT(S)

EFFECTS: POSITIONS_IN_DATAFILE NUMBER_OF_LEVELS TYPE_OF_EFFECT SAVE_
SAMPLES? [EFFECT NESTED]
1 301 cross n   #efeito_do_animal com 1 grupo genético
3   2 cross n   #sexo
RANDOM_RESIDUAL: TYPE PRIOR_DEGREES_OF_BELIEF
homogeneous 10000
RESIDUAL_PRIOR_(CO)VARIANCES
150
RANDOM_GROUP
1
RANDOM_TYPE PRIOR_DEGREES_OF_BELIEF
add_an_upg 1
PEDIGREEFILE: NAME N_ANIMAL N_GENETIC_GROUPS [N_BREEDS]
```



```
animodelrm.pedg 300 1
(CO)VARIANCES
100
```

A.12. Modelo animal via BLUP com efeito normal prior para efeitos fixos

```
# Exemplo 12 - Modelo animal via BLUP com efeito normal prior para efeitos
fixos

BLUP: SOLVE_METHOD CONVERGENCE_CRITERIA [STDERR]

PCG 1e-12 se

DATAFILE NAME N_RECORDS

animodel.dat 240

NUMBER_OF_TRAITS

1

NUMBER_OF_EFFECTS

2

OBSERVATION(S)

4

WEIGHT(S)


EFFECTS: POSITIONS_IN_DATAFILE NUMBER_OF_LEVELS TYPE_OF_EFFECT SAVE_
SAMPLES? [EFFECT NESTED]

1 300 cross n      # efeito_do_animal

3   2 cross n      # sexo

RANDOM_RESIDUAL: TYPE PRIOR_DEGREES_OF_BELIEF

homogeneous 1

RESIDUAL_PRIOR_(CO)VARIANCES

150

RANDOM_GROUP

1

RANDOM_TYPE PRIOR_DEGREES_OF_BELIEF

add_animal 1
```

PEDIGREEFILE: NAME N_ANIMAL N_GENETIC_GROUPS [N_BREEDS]

animodel.ped 300 0

(CO)VARIANCES

100

RANDOM_GROUP

2

RANDOM_TYPE PRIOR_DEGREES_OF_BELIEF

norm_prior

PEDIGREEFILE: NAME N_ANIMAL N_GENETIC_GROUPS [N_BREEDS]

Priors.txt

