Universidade de São Paulo
Escola Superior de Agricultura “Luiz de Queiroz”

Modelos não lineares para dados de contagem longitudinais

Ana Maria Souza de Araujo

Tese apresentada para obtenção do título de Doutor em
Agronomia. Área de concentração: Estatística e Experi-
mentação Agronômica

Piracicaba

2007

Ana Maria Souza de Araujo

Estatística

Modelos não lineares para dados de contagem longitudinais

Orientadora:

Prof^a Dr^a **CLARICE G. B. DEMÉTRIO**

Co-orientador:

Prof. Dr. **DALTON F. DE ANDRADE**

Tese apresentada para obtenção do título de Doutor em
Agronomia. Área de concentração: Estatística e Experi-
mentação Agronômica

Piracicaba

2007

Dados Internacionais de Catalogação na Publicação (CIP)
DIVISÃO DE BIBLIOTECA E DOCUMENTAÇÃO - ESALQ/USP

Araujo, Ana Maria Souza de

Modelos não lineares para dados de contagem longitudinais / Ana Maria Souza de Araujo. -
- Piracicaba, 2007.
197 p. : il.

Tese (doutorado) - - Escola Superior de Agricultura Luiz de Queiroz, 2007.
Bibliografia.

1. Análise de dados longitudinais 2. Dados de contagem 3. Distribuição de Poisson
4. Método da máxima verossimilhança 5. Modelos não lineares I. Título

CDD 519.5

Dedicatória

A **DEUS, Pai, Filho e Espírito Santo**,
fonte de todas as graças e bênçãos derramadas durante o curso de
doutorado e ao longo da minha vida.

À **Maria Santíssima**, minha Mãe do Céu,
constante intercessora junto a Jesus.

Aos meus pais,
Mário Pereira de Araujo (*in memoriam*) e
Maria Estela Sousa de Araujo,
de quem recebi e recebo amor incondicional, doação,
dedicação e apoio constante em todas as fases da minha
vida.

AGRADECIMENTOS

Aos meus orientadores, Prof^ª Clarice Garcia Borges Demétrio e Prof. Dalton Francisco de Andrade, a orientação, o apoio e a amizade.

Aos professores e funcionários do Departamento de Estatística e Matemática Aplicada da Universidade Federal do Ceará, a amizade, o grande incentivo e a confiança em mim depositada.

Aos professores e funcionários do Departamento de Ciências Exatas da ESALQ/USP, a atenção e a amizade.

Ao amigo Padre Carlos Antônio Alcântara de Lima, as orações, a constante presença e a assistência espiritual dada a mim e à minha família.

A Genevile Carife Bergamo, a amizade e o companheirismo ao longo do curso e a convivência alegre durante os tempos de república.

Aos demais amigos e colegas de turma, David José Miquelutti, Denise Nunes Viola, Elisabeth Strapasson, José Carlos Fogo e Pedro Ferreira Filho, a partilha de conhecimentos, a solidariedade nos momentos de angústia e todos os momentos de alegria e descontração.

A todos os amigos dos cursos de mestrado e doutorado do Departamento de Ciências Exatas da ESALQ/USP, que se tornaram a minha família Piracicabana.

Ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) e à Coordenadoria para o Aperfeiçoamento de Pessoal de Nível Superior (CAPES) o apoio financeiro em forma de bolsa.

A todos que, de forma direta ou indireta, contribuíram para a realização deste trabalho.

SUMÁRIO

RESUMO	7
ABSTRACT	8
LISTA DE FIGURAS	9
LISTA DE TABELAS	12
1 INTRODUÇÃO	14
2 DESENVOLVIMENTO	16
2.1 Revisão de Literatura	16
2.1.1 Modelos clássicos para dados longitudinais	16
2.1.2 Modelos generalizados para dados longitudinais	19
2.1.3 Distribuição Poisson Multivariada	22
2.2 Metodologia	25
2.2.1 Material	25
2.2.2 Métodos	27
2.2.2.1 Extensão de Karlis e Meligkotsidou (2005) para acomodar o modelo não linear	27
2.2.2.2 Extensão para delineamento em blocos casualizados com efeitos fixos de bloco	30
2.2.2.3 Extensão para delineamento em blocos casualizados com efeitos aleatórios de bloco	30
2.2.2.4 Modelagem da superdispersão via distribuição Poisson multivariada mista	33
2.2.2.5 Simulação	36
2.2.2.6 Análise dos dados de número de folhas em bromélias	39
2.3 Resultados	40
2.3.1 Simulação	40
2.3.1.1 Delineamento completamente casualizado	40
2.3.1.2 Delineamento em blocos casualizados com efeitos fixos de bloco	41
2.3.1.3 Delineamento em blocos casualizados com efeitos aleatórios de bloco	42
2.3.1.4 Delineamento completamente casualizado com superdispersão	44
2.3.1.5 Delineamento com efeitos fixos de blocos na presença de superdispersão	45
2.3.2 Análise dos dados de número de folhas em bromélias	46
2.3.2.1 Modelo com efeitos fixos de blocos	46
2.3.2.2 Modelo com efeitos aleatórios de blocos	48

	6
3 CONSIDERAÇÕES FINAIS	51
REFERÊNCIAS	53
APÊNDICES	58
ANEXOS	171

RESUMO

Modelos não lineares para dados de contagem longitudinais

Experimentos em que medidas são realizadas repetidamente na mesma unidade experimental são comuns na área agrônômica. As técnicas estatísticas utilizadas para análise de dados desses experimentos são chamadas de análises de medidas repetidas, tendo como caso particular o estudo de dados longitudinais, em que uma mesma variável resposta é observada em várias ocasiões no tempo. Além disso, o comportamento longitudinal pode seguir um padrão não linear, o que ocorre com frequência em estudos de crescimento. Também são comuns experimentos em que a variável resposta refere-se a contagem. Este trabalho abordou a modelagem de dados de contagem, obtidos a partir de experimentos com medidas repetidas ao longo do tempo, em que o comportamento longitudinal da variável resposta é não linear. A distribuição Poisson multivariada, com covariâncias iguais entre as medidas, foi utilizada de forma a considerar a dependência entre os componentes do vetor de observações de medidas repetidas em cada unidade experimental. O modelo proposto por Karlis e Meligkotsidou (2005) foi estendido para dados longitudinais provenientes de experimentos inteiramente casualizados. Modelos para experimentos em blocos casualizados, supondo-se efeitos fixos ou aleatórios para blocos, foram também propostos. A ocorrência de superdispersão foi considerada e modelada através da distribuição Poisson multivariada mista. A estimação dos parâmetros foi realizada através do método de máxima verossimilhança, via algoritmo EM. A metodologia proposta foi aplicada a dados simulados para cada uma das situações estudadas e a um conjunto de dados de um experimento em blocos casualizados em que foram observados o número de folhas de bromélias em seis instantes no tempo. O método mostrou-se eficiente na estimação dos parâmetros para o modelo considerando o delineamento completamente casualizado, inclusive na ocorrência de superdispersão, e delineamento em blocos casualizados com efeito fixo, sem superdispersão e efeito aleatório para blocos. No entanto, a estimação para o modelo que considera efeito fixo para blocos, na presença de superdispersão e para o parâmetro de variância do efeito aleatório para blocos precisa ser aprimorada.

Palavras-chave: Dados de contagem; Dados longitudinais; Modelo não linear; Distribuição Poisson multivariada; Algoritmo EM

ABSTRACT

Non linear models for count longitudinal data

Experiments in which measurements are taken in the same experimental unit are common in agriculture area. The statistical techniques used to analyse data from those experiments are called repeated measurement analysis, and longitudinal study, in which the response variable is observed along the time, is a particular case. The longitudinal behaviour can be non linear, occurring frequently in growth studies. It is also common to have experiments in which the response variable refers to count data. This work approaches the modelling of count data, obtained from experiments with repeated measurements through time, in which the response variable longitudinal behaviour is non linear. The multivariate Poisson distribution, with equal covariances between measurements, was used to consider the dependence between the components of the repeated measurement observation vector in each experimental unit. The Karlis and Meligkotsidou (2005) proposal was extended to longitudinal data obtained from completely randomized. Models for randomized blocks experiments, assuming fixed or random effects for blocks, were also proposed. The occurrence of overdispersion was considered and modelled through mixed multivariate Poisson distribution. The parameter estimation was done using maximum likelihood method, via EM algorithm. The methodology was applied to simulated data for all the cases studied and to a data set from a randomized block experiment in which the number of Bromeliads leaves were observed through six instants in time. The method was efficient to estimate the parameters for the completely randomized experiment, including the occurrence of overdispersion, and for the randomized blocks experiments assuming fixed effect, with no overdispersion, and random effect for blocks. The estimation for the model that considers fixed effect for block, with overdispersion and for the variance parameters of the random effect for blocks must be improved.

Keywords: Count data; longitudinal data; non linear model; Multivariate Poisson distribution; EM Algorithm

LISTA DE FIGURAS

Figura 1 - Número (média de 4 blocos) de folhas em bromélias por tratamento ao longo do tempo	26
Figura 2 - Curva teórica (média de cinco blocos) utilizada na simulação em cada um dos tratamentos	38
Figura 3 - Funções ajustadas e médias de quatro blocos observadas para os cinco tratamentos em cada tempo - Efeitos fixos de blocos	48
Figura 4 - Funções ajustadas e médias de quatro blocos observadas para os cinco tratamentos em cada tempo - Efeitos aleatórios de blocos	49
Figura 5 - Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 1 - Delineamento completamente aleatorizado	181
Figura 6 - Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 2 - Delineamento completamente aleatorizado	181
Figura 7 - Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 3 - Delineamento completamente aleatorizado	182
Figura 8 - Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 4 - Delineamento completamente aleatorizado	182
Figura 9 - Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 1 - Delineamento em blocos casualizados com efeitos fixos de bloco	183
Figura 10 -Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 2 - Delineamento em blocos casualizados com efeitos fixos de bloco	183
Figura 11 -Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 3 - Delineamento em blocos casualizados com efeitos fixos de bloco	184

Figura 12 -Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 4 - Delineamento em blocos casualizados com efeitos fixos de bloco	184
Figura 13 -Histograma representativo das estimativas dos efeitos fixos de blocos	185
Figura 14 -Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 1 - Delineamento em blocos casualizados com efeitos aleatórios de bloco	186
Figura 15 -Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 2 - Delineamento em blocos casualizados com efeitos aleatórios de bloco	186
Figura 16 -Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 3 - Delineamento em blocos casualizados com efeitos aleatórios de bloco	187
Figura 17 -Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 4 - Delineamento em blocos casualizados com efeitos aleatórios de bloco	187
Figura 18 -Histograma representativo das estimativas do parâmetro σ_b^2 - Delineamento em blocos casualizados com efeitos aleatórios de bloco	188
Figura 19 -Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 1 - Delineamento completamente aleatorizado com superdispersão	189
Figura 20 -Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 2 - Delineamento completamente aleatorizado com superdispersão	189
Figura 21 -Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 3 - Delineamento completamente aleatorizado com superdispersão	190
Figura 22 -Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 4 - Delineamento completamente aleatorizado com superdispersão	190

Figura 23 -Histograma representativo das estimativas do parâmetro σ_z^2 - Delineamento completamente aleatorizado com superdispersão	191
Figura 24 -Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 1 - Delineamento em blocos casualizados com efeitos fixos de bloco, com superdispersão	192
Figura 25 -Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 2 - Delineamento em blocos casualizados com efeitos fixos de bloco, com superdispersão	192
Figura 26 -Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 3 - Delineamento em blocos casualizados com efeitos fixos de bloco, com superdispersão	193
Figura 27 -Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 4 - Delineamento em blocos casualizados com efeitos fixos de bloco, com superdispersão	193
Figura 28 -Histograma representativo das estimativas dos efeitos de blocos - Delineamento em blocos casualizados com efeitos fixos de bloco, com superdispersão	194
Figura 29 -Histograma representativo das estimativas do parâmetro σ_z^2 - Delineamento em blocos casualizados com efeitos fixos de bloco, com superdispersão . . .	194

LISTA DE TABELAS

Tabela 1 - Valores médios e variâncias do número de folhas de Bromélia por tempo para as observações de 4 blocos em cada tratamento	26
Tabela 2 - Valores dos parâmetros utilizados no processo de simulação ^(*)	38
Tabela 3 - Médias das estimativas (desvios padrões) obtidos na simulação para o delineamento completamente aleatorizado	40
Tabela 4 - Variância e erro quadrático médio das estimativas obtidas na simulação para o delineamento completamente aleatorizado	40
Tabela 5 - Médias das estimativas (desvios padrões) obtidos na simulação para o modelo com efeitos fixos de blocos*	41
Tabela 6 - Variância e erro quadrático médio das estimativas obtidas na simulação para o modelo com efeitos fixos de blocos	42
Tabela 7 - Médias das estimativas (desvios padrões) obtidos na simulação para o modelo com efeitos aleatórios de blocos	43
Tabela 8 - Variância e erro quadrático médio das estimativas obtidas na simulação para o modelo com efeitos aleatórios de blocos	43
Tabela 9 - Médias das estimativas (desvios padrões) obtidos na simulação para o delineamento completamente aleatorizado na presença de superdispersão	44
Tabela 10 - Variância e erro quadrático médio das estimativas obtidas na simulação para o delineamento completamente aleatorizado na presença de superdispersão	45
Tabela 11 - Médias das estimativas (erros padrões) obtidos na simulação para o modelo com efeitos fixos de blocos na presença de superdispersão	46
Tabela 12 - Variância e erro quadrático médio das estimativas obtidas na simulação para o modelo com efeitos fixos de blocos na presença de superdispersão	46
Tabela 13 - Valores de AIC para cada uma das suposições de igualdades entre parâmetros da curva logística - Efeitos fixos de blocos	47
Tabela 14 - Estimativas dos parâmetros e erros padrões do modelo (2) - Efeitos fixos de blocos	47
Tabela 15 - Valores de AIC para cada uma das suposições de igualdades entre parâmetros da curva logística - Efeitos aleatórios de blocos	48

	13
Tabela 16 -Estimativas dos parâmetros do modelo (2) - Efeitos aleatórios de blocos ⁽¹⁾	49
Tabela 17 -Número total de folhas de bromélias por tratamento e tempo	59
Tabela 18 -Covariâncias (parte superior) e correlações (parte inferior) entre número de folhas de cada tempo, para o tratamento 1	195
Tabela 19 -Covariâncias (parte superior) e correlações (parte inferior) entre número de folhas de cada tempo, para o tratamento 2	195
Tabela 20 -Covariâncias (parte superior) e correlações (parte inferior) entre número de folhas de cada tempo, para o tratamento 3	196
Tabela 21 -Covariâncias (parte superior) e correlações (parte inferior) entre número de folhas de cada tempo, para o tratamento 4	196
Tabela 22 -Covariâncias (parte superior) e correlações (parte inferior) entre número de folhas de cada tempo, para o tratamento 5	197

1 INTRODUÇÃO

Experimentos nos quais uma ou mais variáveis respostas são observadas em cada unidade experimental em mais de uma ocasião e, possivelmente, sob diferentes condições experimentais, são bastante comuns nas áreas agrícola, epidemiológica, biológica, dentre outras. *Medidas repetidas* é o termo usado para descrever os dados obtidos a partir desse tipo de experimento. Como um caso particular de medidas repetidas têm-se os dados longitudinais, para os quais há o interesse em acompanhar o comportamento de uma ou mais variável(is) resposta(s), ao longo de uma dimensão específica, por exemplo, ao longo do tempo.

Quando as variáveis respostas consideradas são contínuas, em geral, supõe-se que seguem distribuição normal, e muita atenção tem sido dada aos procedimentos de estimação. A ênfase é dada à teoria de modelos lineares clássicos, em que, além da normalidade dos dados, pressupõe-se que a relação entre a variável resposta e o fator que representa a medida repetida é linear e que as observações de uma mesma unidade experimental tendem a ser correlacionadas, o que é incorporado através de uma estrutura de covariância apropriada.

No entanto, não raramente, as medidas estudadas são referentes a dados binários, de contagem ou ordinais, e o uso de transformações tem sido adotado a fim de se utilizar a teoria desenvolvida para dados com distribuição normal. Nem sempre, porém, o problema da falta de normalidade é resolvido e, além disso, o emprego de uma teoria que considere a real natureza dos dados produz resultados mais consistentes.

Os modelos lineares generalizados são uma extensão dos modelos lineares clássicos, que supõem que a distribuição do componente aleatório do modelo pertença à família exponencial na forma canônica, com $\mathbf{E}(\mathbf{Y}) = \boldsymbol{\mu}$, e que exista uma função $g(\boldsymbol{\mu})$, monótona, diferenciável, chamada de função de ligação tal que $\boldsymbol{\eta} = g(\boldsymbol{\mu}) = \mathbf{x}^t \boldsymbol{\beta}$.

Introduzida por Nelder e Wedderburn (1972), a teoria dos modelos lineares generalizados levou a um impacto significativo na estatística aplicada, fornecendo uma abordagem unificada a muitos dos procedimentos estatísticos usuais. Inicialmente, de uso restrito, em função dos recursos computacionais limitados na época, atualmente está amplamente difundida. Foi proposta para análise de dados com apenas uma observação por unidade experimental, supondo-se independência entre as observações. Posteriormente, foi estendida

para análise de dados longitudinais, em que a estrutura de covariância entre as medidas de uma mesma unidade deve ser levada em consideração, como também o fato de haver mais de uma fonte de variação aleatória (LIANG; ZEGER, 1986; ZEGER; LIANG, 1986; ZEGER; LIANG; ALBERT, 1988; ZEGER; LIANG, 1992; VONESH; CHINCHILLI, 1997; COSTA, 2003).

Entretanto, a estrutura linear imposta nem sempre é satisfatória para representar o comportamento de uma variável, como ocorre, por exemplo, em estudos de curvas de crescimento, em que a resposta tem comportamento intrinsecamente não linear nos parâmetros. Um aspecto adicional a ser considerado nesse tipo de estudos é que são comuns experimentos em blocos casualizados.

O objetivo deste trabalho consiste na modelagem de dados obtidos a partir de experimentos inteiramente casualizados ou em blocos casualizados, em que a variável resposta é uma contagem ao longo do tempo e tem comportamento não linear. Para incorporar a dependência ao longo do tempo, considera-se a distribuição Poisson multivariada, supondo-se covariâncias iguais entre as contagens repetidas, estendendo-se o modelo proposto por Karlis e Meligkotsidou (2005). Considera-se ainda a ocorrência de superdispersão.

Os objetivos específicos, portanto, são: (i) apresentar revisão da literatura sobre os modelos que tratam do estudo de dados longitudinais, e sobre a distribuição Poisson multivariada; (ii) utilizar a distribuição Poisson multivariada para modelar as dependências entre as diferentes medidas longitudinais em uma mesma unidade experimental, no caso em que um modelo não linear é considerado adequado para representar o comportamento da variável resposta ao longo do tempo; (iii) apresentar uma forma de modelagem da superdispersão, através da distribuição Poisson multivariada mista; (iv) avaliar a metodologia através de simulação e aplicá-la a um conjunto de dados de contagem de folhas em bromélias, ao longo de seis tempos; (vii) discutir os resultados e apresentar sugestões para trabalhos futuros. A implementação computacional da metodologia é feita através do programa R.

2 DESENVOLVIMENTO

2.1 Revisão de Literatura

É comum nas áreas agrícola, biológica, ambiental, dentre outras, haver interesse em modelar o comportamento de variáveis respostas, medidas repetidamente em uma amostra de indivíduos de uma ou mais populações. Pode-se citar como exemplo, a avaliação do impacto de tratamentos tais como herbicidas e fertilizantes no crescimento florestal e na produção, observados nas unidades experimentais em vários instantes no tempo, apresentada em Davidian e Giltinan (2003). Estudos longitudinais constituem um caso especial de medidas repetidas, sendo caracterizados pelo fato de que os dados são coletados em uma dimensão ordenada, por exemplo, o tempo.

Em estudos longitudinais, os dados associados a cada unidade experimental podem ser expressos em termos de um vetor com os valores observados das variáveis respostas em cada instante e de uma matriz com os correspondentes valores das m variáveis explanatórias ou covariáveis. O vetor com as p_i respostas da i -ésima unidade experimental ($i = 1, \dots, n$), chamado de perfil individual, pode ser expresso como

$$\mathbf{Y}_i = (y_{i1}, \dots, y_{ip_i})^t$$

em que y_{ik} representa o valor observado da variável resposta na i -ésima unidade experimental no t -ésimo instante de observação, ($t = 1, \dots, p_i$); $\mathbf{v}^t$ representa o transposto do vetor $\mathbf{v}$. A matriz de variáveis explanatórias, de dimensões $p_i \times m$ pode ser expressa como

$$\mathbf{X}_i = (\mathbf{x}_{i1}, \dots, \mathbf{x}_{im})$$

em que $\mathbf{x}_{ij} = (x_{ij1}, \dots, x_{ijp_i})^t$ com x_{ijt} sendo o valor da j -ésima variável explanatória ($j = 1, \dots, m$) para a i -ésima unidade experimental no t -ésimo instante de observação.

2.1.1 Modelos clássicos para dados longitudinais

Métodos de análise de dados longitudinais com distribuição gaussiana estão bastante desenvolvidos, havendo atualmente, uma ampla classe de modelos disponíveis. Podem-se citar como clássicos na área os trabalhos publicados por Wishart (1938), Box (1950), Geisser e Greenhouse (1958), Pothoff e Roy (1964), Rao (1959, 1965 - 67), Elston e Grizzle (1962)

e Grizzle e Allen (1969). Métodos para o ajuste de curvas de crescimento a dados de medidas repetidas são discutidos em Rao (1965), Grizzle e Allen (1969), Fearn (1975), Hui (1984), dentre outros, enquanto que Harville (1977) e Laird e Ware (1982) desenvolveram modelos em que observações repetidas para uma mesma unidade experimental são supostas ter um componente aleatório comum, permitindo a modelagem da estrutura de covariância entre elas. São chamados de modelos de efeitos mistos ou modelos de efeitos aleatórios.

De uma forma geral, supõe-se:

$$\mathbf{Y}_i = f(\mathbf{X}_i, \boldsymbol{\beta}, \mathbf{b}_i) + \boldsymbol{\varepsilon}_i, \quad i = 1, \dots, n, \quad (1)$$

em que f é uma função vetor conveniente, $\boldsymbol{\beta} = (\beta_1, \dots, \beta_m)^t$ é o vetor de parâmetros de efeitos fixos, $\mathbf{b}_i = (b_{i1}, \dots, b_{iq})^t$ é o vetor de efeitos aleatórios com vetor de médias $\mathbf{0}$ e matriz de variâncias e covariâncias $\mathbf{D}$ e $\boldsymbol{\varepsilon}_i = (\varepsilon_{i1}, \dots, \varepsilon_{ip_i})^t$ é um vetor de perturbações aleatórias com média $\mathbf{0}$ e matriz de covariâncias $\mathbf{G}_i$. As matrizes de covariâncias intra-unidades, $\boldsymbol{\Sigma}_i$, são funções de $\mathbf{X}_i$, $\boldsymbol{\beta}$, $\mathbf{D}$ e $\mathbf{G}_i$. No entanto, muitos problemas práticos podem ser tratados através de uma simplificação de (1) dada por

$$\mathbf{Y}_i = f(\mathbf{X}_i, \boldsymbol{\beta}) + \boldsymbol{\varepsilon}_i,$$

ou seja, $\boldsymbol{\Sigma}_i = \mathbf{G}_i$, tal que as matrizes de covariâncias intra-unidades são modeladas independentemente dos parâmetros de locação.

Tratando-se do modelo linear misto clássico, pode-se considerar o de dois estágios, proposto por Laird e Ware (1982) e a equação (1) pode ser escrita como

$$\mathbf{Y}_i = \mathbf{X}_i \boldsymbol{\beta} + \mathbf{Z}_i \mathbf{b}_i + \boldsymbol{\varepsilon}_i$$

em que $\boldsymbol{\beta}$ é um vetor de dimensões $m \times 1$ de parâmetros desconhecidos, $\mathbf{X}_i$ é uma matriz de delineamento conhecida, de dimensões $p_i \times m$, associada ao i -ésimo indivíduo, $\mathbf{b}_i$ é um vetor de efeitos aleatórios individuais desconhecidos, de dimensões $q \times 1$, $\mathbf{Z}_i$ é uma matriz de delineamento conhecida, de dimensões $p_i \times q$ e $\boldsymbol{\varepsilon}_i$ um vetor de perturbações aleatórias, seguindo distribuição $N(\mathbf{0}, \mathbf{G}_i)$, sendo $\mathbf{G}_i$ uma matriz positiva definida de covariâncias, de dimensões $p_i \times p_i$. No primeiro estágio, são considerados fixos $\boldsymbol{\beta}$ e $\mathbf{b}_i$ e os $\boldsymbol{\varepsilon}_i$'s são supostos independentes, tais que:

$$\mathbf{E}(\mathbf{Y}_i \mid \mathbf{b}_i) = \mathbf{X}_i \boldsymbol{\beta} + \mathbf{Z}_i \mathbf{b}_i \text{ e } \mathbf{V}(\mathbf{Y}_i \mid \mathbf{b}_i) = \mathbf{G}_i.$$

No segundo estágio, os $\mathbf{b}_i$'s têm distribuição $N(\mathbf{0}, \mathbf{D})$, independentes entre si e independentes dos $\boldsymbol{\varepsilon}_i$'s. $\mathbf{D}$ é uma matriz de covariâncias positiva definida, de dimensões $q \times q$. O vetor de parâmetros $\boldsymbol{\beta}$ continua sendo considerado fixo. Assim, marginalmente, os $\mathbf{Y}_i$ são independentes, e

$$\mathbf{E}(\mathbf{Y}_i) = \mathbf{X}_i \boldsymbol{\beta} \text{ e } \boldsymbol{\Sigma}_i = \mathbf{V}(\mathbf{Y}_i) = \mathbf{Z}_i \mathbf{D} \mathbf{Z}_i^t + \mathbf{G}_i.$$

Quando $\mathbf{G}_i = \sigma^2 \mathbf{I}$ e $\mathbf{Z}_i = \mathbf{0}$, tem-se o modelo linear padrão.

A estimação dos parâmetros em modelos mistos pode ser realizada através do algoritmo EM (*“Expectation-Maximization”*), desenvolvido por Dempster, Laird e Rubin (1977), para ser aplicado quando as observações podem ser vistas como dados incompletos. Cada iteração é constituída de um passo de cálculo da esperança (Passo **E**) e um segundo passo para maximização da função (Passo **M**). Os dados, chamados de incompletos, são constituídos de dois espaços, um formado pelos dados observados, $\mathbf{y}$ e outro não observável diretamente, $\mathbf{b}$, mas somente através de $\mathbf{y}$. Em modelos mistos, os efeitos aleatórios são tratados como dados não observados ($\mathbf{b}$).

Em linhas gerais, o algoritmo pode ser descrito como a seguir. Seja $f(\mathbf{b} | \boldsymbol{\Phi})$ a família de densidades de $\mathbf{b}$ e $g(\mathbf{y} | \boldsymbol{\Phi})$ a família de densidades de $\mathbf{Y}$, ambas dependendo do vetor de parâmetros de dimensões $r \times 1$, $\boldsymbol{\Phi} \in \Omega$. Seja, também,

$$Q(\boldsymbol{\theta}' | \boldsymbol{\Phi}) = \mathbf{E}(\log f(\mathbf{b} | \boldsymbol{\Phi}') | \mathbf{y}, \boldsymbol{\Phi}),$$

supondo que existam todos os pares $(\boldsymbol{\Phi}, \boldsymbol{\Phi}')$. A realização de várias repetições do algoritmo:

- (i) Passo E: Calcular $Q(\boldsymbol{\Phi} | \boldsymbol{\Phi}^{(p)})$, em que $\boldsymbol{\Phi}^{(p)}$ é o valor de $\boldsymbol{\Phi}$ após p iterações do algoritmo;
- (ii) Passo M: Encontrar $\boldsymbol{\theta}^{(p+1)}$ que maximize $Q(\boldsymbol{\Phi} | \boldsymbol{\Phi}^{(p)})$,

leva ao valor de $\boldsymbol{\Phi}$ que maximiza $L(\boldsymbol{\Phi} | \mathbf{y}) = \log(g(\mathbf{y} | \boldsymbol{\Phi}))$.

Quando $f(\mathbf{b} | \boldsymbol{\Phi})$ tem a forma da família exponencial, ou seja,

$$f(\mathbf{b} | \boldsymbol{\Phi}) = d(\mathbf{b})c(\boldsymbol{\Phi}) \exp(\boldsymbol{\Phi}^t \mathbf{t}(\mathbf{b})),$$

em que $\mathbf{t}(\mathbf{b})$ é um vetor de dimensões $r \times 1$ de estatísticas suficientes, o processo é dado por:

- (iii) Passo E: Encontrar $\mathbf{t}^{(p)} = \mathbf{E}(\mathbf{t}(\mathbf{b}) | \mathbf{y}, \boldsymbol{\Phi}^{(p)})$;

(iv) Passo M: Encontrar $\Phi^{(p+1)}$ que é solução da equação $\mathbf{E}(\mathbf{t}(\mathbf{b}) \mid \Phi^{(p)}) = \mathbf{t}^{(p)}$.

Se a função densidade de probabilidade do vetor de dados observados, $\mathbf{y}$, é dada por $g(\mathbf{y} \mid \boldsymbol{\theta})$ e o vetor de parâmetros $\boldsymbol{\theta}$ tem uma distribuição pertencente à família de densidades $h(\boldsymbol{\theta} \mid \Phi)$, pode-se considerar como vetor $\mathbf{b} = (\mathbf{y}, \boldsymbol{\theta})$. Neste caso, o passo M consiste em encontrar

$$\mathbf{E}(\log f(\mathbf{b} \mid \Phi^{(p)}) \mid \mathbf{y}, \boldsymbol{\theta}) = \mathbf{E}(\log(g(\mathbf{y} \mid \boldsymbol{\theta}))) + \mathbf{E}(\log(h(\boldsymbol{\theta} \mid \Phi^{(p)}))).$$

No passo M, encontra-se Φ tal que essa esperança seja maximizada. Quando $h(\boldsymbol{\theta} \mid \Phi^{(p)})$ pertence à família exponencial, com estatística suficiente $\mathbf{t}(\boldsymbol{\theta})$, os passos E e M tornam-se equivalentes aos descritos nos itens (iii) e (iv).

Em muitas situações, um modelo intrinsecamente não linear nos parâmetros parece ser mais adequado para representar o comportamento da variável sob estudo. Nesse caso, o modelo é o mesmo apresentado em (1), sendo f uma função não linear específica de $(\mathbf{X}_i, \boldsymbol{\beta}, \mathbf{b}_i)$.

Estudo detalhado de modelos para dados contínuos longitudinais estão em Davidian e Giltinan (1995), Vonesh e Chinchilli (1997), Diggle, Singer e Andrade (2000) e Liang e Zeger (2002). O caso específico de modelos não lineares para análise de dados longitudinais provenientes de experimentos em blocos casualizados com variável resposta seguindo distribuição normal é estudado em Ogliari (1998) e Ogliari e Andrade (2001). Os blocos, tratados como efeitos fixos ou aleatórios, são modelados de formas linear e não linear.

2.1.2 Modelos generalizados para dados longitudinais

Em modelos de regressão para dados com uma observação por unidade experimental, os modelos lineares generalizados (NELDER; WEDDERBURN, 1972) e a teoria de quase-verossimilhança, proposta por Wedderburn (1974) e extensivamente examinada por McCullagh (1983) e McCullagh e Nelder (1983), ampliaram a aplicação dos modelos lineares para dados com distribuição normal para uma classe mais ampla. A abordagem de quase-verossimilhança não exige especificação da distribuição da variável dependente, apenas assume que a variância é uma função conhecida de sua esperança.

Os modelos logísticos de efeitos aleatórios para dados binários (COX, 1970;

KOCH et al., 1977; STIRATELLI et al., 1984; OCHI; PRENTICE, 1984; ANDERSON; AITKIN, 1985; ZEGER; LIANG; SELF, 1985) estão entre os primeiros modelos para medidas repetidas. No entanto, esses modelos, com exceção do modelo de Stiratelli et al. (1984), poderiam ser usados apenas na presença também de covariáveis sem dependência de tempo.

Posteriormente, Liang e Zeger (1986) fazem a suposição de que a densidade marginal de Y_{it} , $t = 1, \dots, p_i$, pertence à família exponencial na forma canônica e apresentam as equações de estimação em que supõem que as observações repetidas de uma unidade experimental são independentes. Em seguida, tratam da estimação do vetor de parâmetros β , utilizando as equações de estimação generalizadas (GEE), que têm como pressuposição em sua estrutura, a dependência entre as observações de uma mesma unidade experimental através de uma matriz de correlação chamada *matriz de correlação de trabalho*.

Metodologia menos restritiva foi proposta por Zeger e Liang (1986), utilizando a abordagem da quase-verossimilhança, ou seja, exigindo apenas que uma função da esperança marginal da variável dependente seja uma função linear das covariáveis e que a variância seja uma função conhecida da média. As GEE para análise de dados longitudinais também são utilizadas em Zeger, Liang e Albert (1988), no caso em que os parâmetros do modelo linear generalizado variam com o indivíduo, segundo alguma distribuição de mistura. Introduz os termos “Subject-Specific”, para se referir à situação em que a heterogeneidade é especificamente modelada, e “Population-Averaged”, quando a resposta é modelada em função de covariáveis, sem levar em consideração a heterogeneidade indivíduo a indivíduo.

O caso particular do modelo não linear para medidas repetidas, de acordo com Vonesh e Chinchilli (1997), pode ser representado por:

$$\mathbf{Y}_i | \mathbf{b}_i = f(\mathbf{X}_i, \beta_i) + \mathbf{B}_{i2}(\beta) \mathbf{b}_{i2} + \mathbf{\Lambda}_i^{1/2}(\beta_i, \gamma) \boldsymbol{\varepsilon}_i, \quad i = 1, \dots, n, \quad (2)$$

sendo (i) $\beta_i = g(\mathbf{a}_i, \beta) + \mathbf{B}_{i1} \mathbf{b}_{i1}$; (ii) $f(\mathbf{X}_i, \beta_i)$ uma função vetor de dimensões $p_i \times 1$, que é possivelmente não linear em β_i ; (iii) $\mathbf{b}_{i1}$ um vetor $v_1 \times 1$ de efeitos aleatórios independentes e identicamente distribuídos, com vetor de médias $\mathbf{0}$ e matriz de variâncias e covariâncias Ψ_{11} ; (iv) $\mathbf{b}_{i2}$ um vetor $v_2 \times 1$ de efeitos aleatórios independentes e identicamente distribuídos, com vetor de médias $\mathbf{0}$ e matriz de variâncias e covariâncias Ψ_{22} ; (v) $\text{Cov}(\mathbf{b}_{i1}, \mathbf{b}_{i2}) = \Psi_{12}$; (vi) $\boldsymbol{\varepsilon}_i$ é um vetor de perturbações aleatórias intra-unidade da amostra, independente de $\mathbf{b}_i$, de

dimensões $p_i \times 1$, composto por ε_{ik} que são independentes e identicamente distribuídos, cada um com média 0 e variância σ^2 ; (vii) $\mathbf{B}_{i2}(\boldsymbol{\beta})$ é uma matriz $p_i \times v_2$, consistindo de funções conhecidas de $\boldsymbol{\beta}$, $\mathbf{a}_i$ e $\mathbf{X}_i$; (viii) $g(\mathbf{a}_i, \boldsymbol{\beta})$ é uma vetor $r \times 1$, função de $\mathbf{a}_i$ ($q \times 1$) e $\boldsymbol{\beta}$ ($s \times 1$) e (ix) $\sigma^2 \boldsymbol{\Lambda}_i(\boldsymbol{\beta}_i, \boldsymbol{\gamma})$ é uma matriz de covariância intra-unidade experimental, que depende de parâmetros de variância desconhecidos $[\sigma^2 \boldsymbol{\gamma}^t]$ e possivelmente de alguma função de $\boldsymbol{\beta}_i$ (ou, de forma mais geral, de $\boldsymbol{\beta}$ e $\mathbf{b}_i$).

Essa forma de apresentar o modelo em (2) tem como vantagem permitir a identificação dos efeitos que são estritamente lineares, o que pode simplificar os procedimentos de estimação. Além disso, observa-se que a única suposição a respeito dos efeitos aleatórios é que tenham média $\mathbf{0}$ e variâncias finitas, não havendo restrições quanto à distribuição, permitindo que seja usado para dados discretos ou contínuos. Possibilita, também, a escolha da estrutura de covariância a ser utilizada.

Pode-se ver também que diferentes definições das matrizes $\mathbf{B}_{i1}$ e $\mathbf{B}_{i2}(\boldsymbol{\beta})$ levam a diferentes classes de modelos. Se $\mathbf{B}_{i1} = \mathbf{0}$ e $\mathbf{B}_{i2}(\boldsymbol{\beta}) = \mathbf{0}$ têm-se modelos não lineares marginais de efeitos fixos; se apenas $\mathbf{B}_{i1} = \mathbf{0}$ ou se ambas são diferentes de $\mathbf{0}$ tem-se uma classe geral de modelos não lineares de efeitos mistos.

A obtenção das estimativas dos parâmetros através do método de máxima verossimilhança é realizada pela maximização do logaritmo da função de verossimilhança conjunta

$$L(\boldsymbol{\beta}, \boldsymbol{\theta} \mid \mathbf{Y}_1, \dots, \mathbf{Y}_n) = \sum_{i=1}^n \log \int \pi(\mathbf{Y}_i \mid \mathbf{b}_i) d\mathbf{D}_v(\mathbf{b}_i \mid \mathbf{0}, \boldsymbol{\Psi}),$$

em que $\pi(\mathbf{Y}_i \mid \mathbf{b}_i)$ é a função densidade de probabilidade condicional de $\mathbf{Y}_i$ dado $\mathbf{b}_i$ e $\mathbf{D}_v(\mathbf{b}_i \mid \mathbf{0}, \boldsymbol{\Psi})$ é a função densidade de probabilidade do vetor de efeitos aleatórios $\mathbf{b}_i$. Em geral, não há soluções analíticas para esse sistema, sendo necessárias aproximações numéricas.

Os métodos descritos são gerais, não exigindo a especificação de alguma distribuição em particular, sendo, portanto, válidos para dados seguindo qualquer distribuição de probabilidade, discreta ou contínua. No entanto, é necessário que a estrutura adotada de covariâncias intra-unidade de investigação seja coerente com a distribuição dos dados. Por exemplo, quando se supõe distribuição gaussiana para os dados, a variância independe da média, o que não acontece com dados seguindo distribuição Poisson. Esse fato deve ser levado em consideração no processo de modelagem.

2.1.3 Distribuição Poisson Multivariada

Quando os dados se referem a contagens observadas longitudinalmente, sendo $\mathbf{Y}_i = (Y_{i1}, Y_{i1}, \dots, Y_{im})^t$, o vetor de observações associados a elas, de dimensões $m \times 1$, em que cada uma de suas componentes é resultante de uma medida realizada na mesma unidade de observação, espera-se que haja correlação entre elas. Uma forma de modelar a dependência existente entre as observações é supor que existe um conjunto de variáveis latentes W_{ir} , $r = 1, \dots, k$, independentes, seguindo distribuição Poisson, isto é, $W_{ir} \sim P(\theta_r)$, e que os componentes de $\mathbf{Y}_i$ são combinações lineares delas. Segundo essa estrutura, $\mathbf{Y}_i$ tem distribuição Poisson m-variada.

Definindo a distribuição Poisson multivariada, seja $\mathbf{A} = [\mathbf{A}_1, \dots, \mathbf{A}_q]$ uma matriz de dimensões $m \times k$, $m \leq k$, formada por q submatrizes, cada uma delas de dimensões $m \times \binom{m}{j}$, $j = 1, \dots, q$, sendo que cada coluna de $\mathbf{A}_j$ tem exatamente j 1's e $(m-j)$ 0's, e não existe coluna duplicada. O vetor $\mathbf{Y}_i = \mathbf{A}\mathbf{W}_i$, em que $\mathbf{W}_i = (W_{i1}, W_{i2}, \dots, W_{ir})^t$, tem distribuição Poisson multivariada, com $\mathbf{E}(\mathbf{Y}_i) = \mathbf{A}\boldsymbol{\theta}$ e $\mathbf{V}(\mathbf{Y}_i) = \mathbf{A}\boldsymbol{\Sigma}\mathbf{A}^t$, sendo $\boldsymbol{\theta} = \mathbf{E}(\mathbf{W}_i) = [\theta_1, \dots, \theta_k]^t$ e $\boldsymbol{\Sigma} = \text{diag}(\theta_1, \dots, \theta_k)$. Dessa forma, a estrutura de dependência dos elementos do vetor $\mathbf{Y}_i$ é determinada pela matriz $\mathbf{A}$ e cada elemento de $\mathbf{Y}_i$ segue marginalmente uma distribuição Poisson univariada.

Assim, por exemplo, se $\mathbf{A} = [A_1 : A_4]_{4 \times 5}$, então

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 & 0 & \vdots & 1 \\ 0 & 1 & 0 & 0 & \vdots & 1 \\ 0 & 0 & 1 & 0 & \vdots & 1 \\ 0 & 0 & 0 & 1 & \vdots & 1 \end{bmatrix} \quad (3)$$

e $\mathbf{W}_i = [W_{i1} \ W_{i2} \ W_{i3} \ W_{i4} \ W_{i5}]^t$, em que $W_{ir} \sim P(\theta_r)$, $r = 1, 2, \dots, 5$, o vetor $\mathbf{Y}_i = [Y_{i1}, \dots, Y_{i4}]^t = \mathbf{A}\mathbf{W}_i = (W_{i1} + W_{i5}, W_{i2} + W_{i5}, W_{i3} + W_{i5}, W_{i4} + W_{i5})^t$ tem distribuição Poisson tetravariada, com parâmetros $(\theta_1, \theta_2, \theta_3, \theta_4, \theta_5)$, denotada por

$\mathbf{Y}_i \sim P_{4P}(\theta_1, \theta_2, \theta_3, \theta_4, \theta_5)$. Tem-se que $\mathbf{E}(\mathbf{Y}_i) = [\theta_1 + \theta_5, \theta_2 + \theta_5, \theta_3 + \theta_5, \theta_4 + \theta_5]^t$ e

$$V(\mathbf{Y}_i) = \begin{bmatrix} \theta_1 + \theta_5 & \theta_5 & \theta_5 & \theta_5 \\ \theta_5 & \theta_2 + \theta_5 & \theta_5 & \theta_5 \\ \theta_5 & \theta_5 & \theta_3 + \theta_5 & \theta_5 \\ \theta_5 & \theta_5 & \theta_5 & \theta_4 + \theta_5 \end{bmatrix}, \quad (4)$$

o que leva a covariâncias iguais entre as observações longitudinais. Pode-se ver que a variável latente W_{i5} , comum a todas as combinações lineares, é quem modela a dependência entre os componentes de $\mathbf{Y}_i$.

A fim de definir a função de probabilidade conjunta da distribuição Poisson multivariada, seja $g : N^k \rightarrow N^m$, $k \geq m$, tal que $\mathbf{Y}_i = g(\mathbf{W}_i) = \mathbf{A}\mathbf{W}_i$, em que $\mathbf{A}$ e $\mathbf{W}_i$ são como definidas anteriormente e $\mathbf{Y}_i \sim P_{mP}(\theta_1, \dots, \theta_k)$. A função de probabilidade conjunta de $\mathbf{Y}_i = (Y_1, Y_2, \dots, Y_m)^t$ é dada por:

$$P(\mathbf{Y}_i = \mathbf{y}_i) = \sum_{\mathbf{w}_i \in g^{-1}(\mathbf{y}_i)} P(\mathbf{W}_i = \mathbf{w}_i) = \sum_{\mathbf{w}_i \in g^{-1}(\mathbf{y}_i)} \prod_{r=1}^k P(w_{ir}, \theta_r),$$

visto que os elementos de $\mathbf{W}_i$ são independentes.

A inclusão de covariáveis nesse modelo pode ser feita usando-se $\log(\theta_r) = \mathbf{z}_r^t \boldsymbol{\beta}$, em que $\mathbf{z}_r$ é um vetor de covariáveis de dimensões $l \times 1$ e $\boldsymbol{\beta} = (\beta_1, \dots, \beta_l)$ é um vetor de coeficientes de regressão associados ao r -ésimo parâmetro. A estimação dos parâmetros pode ser feita por máxima verossimilhança usando o algoritmo EM (KARLIS; MELIGKOTSIDOU, 2005), em dois estágios, com uma combinação de máxima verossimilhança e mínimos quadrados generalizados (HO; SINGER, 2001) ou por métodos Bayesianos (TSIONAS, 2001). Aplicações foram feitas a dados de contagens de quatro diferentes tipos de crime em 49 prefeituras gregas no ano de 1997 (KARLIS; MELIGKOTSIDOU, 2005) e a dados simulados e dados de destruição de florestas no continente europeu (TSIONAS, 2001).

No processo de estimação apresentado em Karlis e Meligkotsidou (2005), consideram-se como dados observados os vetores $\mathbf{Y}_i$ e como não observados os vetores de variáveis latentes $\mathbf{W}_i$. No passo E, calculam-se as esperanças condicionais (s_{ir}) das variáveis latentes W_{ir} condicionadas aos dados observados e às estimativas correntes dos parâmetros.

As esperanças condicionais são dadas por

$$s_{ir} = \mathbf{E}(W_{ir} \mid \mathbf{Y}_i, t_i, \boldsymbol{\theta}^{(p)}) = \frac{\sum_{\mathbf{w}_i \in g^{-1}(\mathbf{y}_i)} w_{ir} \prod_{r=1}^k \mathbf{P}(w_{ir} \mid \theta_{ij}^{(p)})}{\mathbf{P}(\mathbf{y}_i \mid \theta_{ir}^{(p)} t_i)}.$$

No passo M, os valores esperados s_{ir} são tratados como verdadeiras observações de W_{ir} e maximiza-se a verossimilhança dos dados completos. Quando os parâmetros de covariância são funções de covariáveis, atualiza-se o vetor $\boldsymbol{\beta}$ ajustando uma regressão Poisson (por Modelos Lineares Generalizados), usando os s'_{ir} s como variáveis respostas e as variáveis explanatórias $\mathbf{z}_{ir}$, $i = 1, \dots, n$. Os passos E e M são repetidos até a convergência do processo, segundo algum critério pré-estabelecido.

O modelo apresentado tem como restrição que $\mathbf{E}(\mathbf{Y}_i) = \mathbf{V}(\mathbf{Y}_i)$, o que nem sempre ocorre na prática, sendo mais comum a obtenção de $\mathbf{V}(\mathbf{Y}_i) > \mathbf{E}(\mathbf{Y}_i)$, o que é chamado de superdispersão. É possível ocorrer, também, o caso de $\mathbf{V}(\mathbf{Y}_i) < \mathbf{E}(\mathbf{Y}_i)$, chamado de subdispersão, no entanto, menos comum.

A superdispersão pode resultar de dois processos diferentes, isto é, por variabilidade da média da distribuição Poisson e/ou por excesso de zeros, em geral, estruturais.

Uma forma de modelar a superdispersão é através das distribuições Poisson mistas, por exemplo, a distribuição Poisson mista multivariada que, segundo Karlis e Xekalaki (2005), podem ser do tipo 1 ou tipo 2. A distribuição Poisson multivariada mista do tipo 1 é aquela em que os parâmetros da distribuição Poisson multivariada são proporcionais a alguma variável aleatória a , com densidade $g(a)$. No caso bivariado, a função de probabilidade conjunta de uma distribuição Poisson mista é dada por:

$$P_g(y_1, y_2) = \int_a P_2(y_1, y_2, a\theta_1, a\theta_2, a\theta_3) g(a) da,$$

em que $P_2(y_1, y_2, a\theta_1, a\theta_2, a\theta_3)$ é a função distribuição da Poisson bivariada, com vetor de parâmetros $(a\theta_1, a\theta_2, a\theta_3)^t$. Quando os parâmetros da distribuição Poisson multivariada são variáveis aleatórias, tem-se a distribuição Poisson multivariada mista do tipo 2. Uma revisão de literatura em distribuições Poisson mistas é feita em Karlis e Xekalaki (2005).

No caso particular da superdispersão resultante de excessos de zero, Li et al. (1999) apresentam os modelos Poisson multivariados inflacionados de zeros, enquanto que Walhin (2001) propõe diversas alternativas para o caso de Poisson bivariado.

Aplicações usando a Poisson bivariada inflacionada de zeros são feitas em Wang et al. (2003) que estuda um conjunto de dados referentes a número de acidentes de trabalho após intervenções realizadas no setor de limpeza de um hospital de clínicas.

Outra forma de modelar superdispersão foi proposta por Karlis e Ntzoufras (2003) e Karlis e Ntzoufras (2005) que consideram modelos de regressão de Poisson bivariada com diagonal inflacionada, para modelar superdispersão ocasionada por excesso de valores iguais das variáveis Y_1 e Y_2 . Karlis e Ntzoufras (2003) ilustram a proposta modelando resultados de jogos de futebol na série A do campeonato italiano na temporada 1991-1992, em que as variáveis observadas são os números de gols de cada equipe, havendo um grande número de empates.

2.2 Metodologia

2.2.1 Material

Os dados a serem utilizados neste trabalho são provenientes de um experimento em blocos casualizados com 4 blocos, realizado no período de julho de 1997 a outubro de 1998, conduzido no Departamento de Fitotecnia da Escola Superior de Agricultura “Luiz de Queiroz”, ESALQ/USP, Piracicaba-SP. Nele desejava-se avaliar substratos que pudessem ser utilizados na substituição de xaxim no cultivo de Bromélias (*Aechmea fasciata*) em vaso. Uma variável avaliada foi o número total de folhas de 8 plantas de Bromélias, em 6 instantes no tempo que correspondem a 5, 173, 229, 285, 341 e 435 dias após o plantio (ver Tabela 17, no Apêndice A). Os 15 substratos usados eram combinações de algum material, sempre na companhia de turfa e perlita, em três diferentes proporções desses materiais (2:7:1), (5:4:1), (8:1:1). Aqui, da mesma forma que em Xavier (2000), serão considerados cinco substratos como tratamentos, todos na proporção (5:4:1), quais sejam:

Tratamento 1: Substrato 1 - Casca de *Pinus* + turfa + perlita;

Tratamento 2: Substrato 2 - Casca de *Eucalyptos* + turfa + perlita;

Tratamento 3: Substrato 3 - Coxim + turfa + perlita;

Tratamento 4: Substrato 4 - Fibra de coco + turfa + perlita;

Tratamento 5: Substrato 5 - Xaxim + turfa + perlita.

Os valores médios do número de folhas em cada um dos tempos e suas res-

pectivas variâncias estão apresentados na Tabela 1, enquanto que o comportamento médio por tratamento pode ser observado na Figura 1. Vê-se que há uma tendência não linear ao longo do tempo, o que é comum em experimentos desse tipo. Por outro lado, nota-se que as variâncias observadas são menores do que as médias, indicando subdispersão. Observa-se também que, de uma forma geral, as variâncias aumentam ao longo do tempo, ocorrendo algumas exceções.

Tabela 1 - Valores médios e variâncias do número de folhas de Bromélia por tempo para as observações de 4 blocos em cada tratamento

Tempo (dias após plantio)	Trat1		Trat2		Trat3		Trat4		Trat5	
	Média	Var.	Média	Var.	Média	Var.	Média	Var.	Média	Var.
5	52,50	9,67	52,25	7,58	52,50	3,00	51,75	4,92	53,75	1,58
173	72,00	2,67	60,00	12,00	64,25	9,58	65,50	12,33	75,00	42,00
229	92,00	10,00	82,00	22,00	86,75	18,25	84,25	22,92	98,75	46,25
285	113,00	18,00	99,00	18,00	105,75	42,25	103,25	5,58	109,25	2,25
341	130,50	27,00	115,25	8,92	119,25	26,25	118,25	38,25	127,00	31,33
435	137,75	42,92	122,75	12,92	126,50	21,67	125,50	27,00	133,00	22,67

Figura 1 - Número (média de 4 blocos) de folhas em bromélias por tratamento ao longo do tempo

2.2.2 Métodos

2.2.2.1 Extensão de Karlis e Meligkotsidou (2005) para acomodar o modelo não linear

Seja Y_{ijt} a variável aleatória correspondente a uma observação feita na unidade experimental i , $i = 1, \dots, n$, do tratamento j , $j = 1, \dots, k$, no tempo t , $t = 1, \dots, T$, resultante de um experimento no delineamento completamente casualizado. Supondo-se que a variável aleatória é uma contagem e que existe uma dependência entre as observações feitas ao longo do tempo, é natural considerar um modelo Poisson multivariado para $\mathbf{Y}_{ij} = (Y_{ij1}, Y_{ij2}, \dots, Y_{ij,T+1})$. Uma forma de se introduzir a dependência existente entre Y_{ijt} e $Y_{ijt'}$, $t, t' = 1, \dots, T$ é supor que Y_{ijt} é uma combinação linear de duas variáveis latentes independentes, como visto na seção 2.3.1, isto é, $Y_{ijt} = W_{ijt} + W_{ij,T+1}$, em que $W_{ijt} \sim P(\theta_{jt})$ e que $W_{ij,T+1} \sim P(\theta_{j,T+1})$ é comum a todas as observações feitas ao longo do tempo para a i -ésima repetição do j -ésimo no tratamento. Portanto, $\mathbf{Y}_{ij}$ tem distribuição Poisson T -variada com vetor de parâmetros $\boldsymbol{\theta}_j = (\theta_{j1}, \theta_{j2}, \dots, \theta_{j,T+1})$, isto é, $\mathbf{Y}_{ij} \sim P_T(\theta_{j1}, \theta_{j2}, \dots, \theta_{jT}, \theta_{j,T+1})$, com função de probabilidade dada por:

$$\begin{aligned}
 P(\mathbf{Y}_{ij}) &= P(Y_{ij1} = y_{ij1}, Y_{ij2} = y_{ij2}, \dots, Y_{ijT} = y_{ijT}) = \\
 &= \sum_{w_{ij,T+1}=0}^{\min(\mathbf{y}_{ij})} P(W_{ij,T+1} = w_{ij,T+1}, W_{ij1} = y_{ij1} - w_{ij,T+1}, W_{ij2} = y_{ij2} - w_{ij,T+1}, \dots, \\
 &\quad \dots, W_{ijt} = y_{ijt} - w_{ij,T+1}) \\
 &= \sum_{w_{ij,T+1}=0}^{\min(\mathbf{y}_{ij})} \left[P(W_{ij,T+1} = w_{ij,T+1}) \prod_{t=1}^T (P(W_{ijt} = y_{ijt} - w_{ij,T+1})) \right] \\
 &= \sum_{w_{ij,T+1}=0}^{\min(\mathbf{y}_{ij})} \left[\frac{e^{-\theta_{j,T+1}} \theta_{j,T+1}^{w_{ij,T+1}}}{w_{ij,T+1}!} \prod_{t=1}^T \left(\frac{e^{-\theta_{jt}} \theta_{jt}^{y_{ijt} - w_{ij,T+1}}}{(y_{ijt} - w_{ij,T+1})!} \right) \right] \\
 &= e^{(-\sum_{m=1}^{T+1} \theta_{jm})} \left(\prod_{t=1}^T \frac{\theta_{jt}^{y_{ijt}}}{y_{ijt}!} \right) \sum_{w_{ij,T+1}=0}^{\min(\mathbf{y}_{ij})} \left\{ \left(\frac{\theta_{j,T+1}}{\prod_{t=1}^T \theta_{jt}} \right)^{w_{ij,T+1}} \frac{1}{w_{ij,T+1}!} \prod_{t=1}^T \left[\binom{y_{ijt}}{w_{ij,T+1}} w_{ij,T+1}! \right] \right\}.
 \end{aligned}$$

Note que, marginalmente, as variáveis aleatórias Y_{ijt} têm distribuição $P(\lambda_{jt})$, e, portanto, $\mathbf{E}(Y_{ijt}) = \mathbf{V}(Y_{ijt}) = \lambda_{jt} = \theta_{jt} + \theta_{j,T+1}$ e que $\text{Cov}(Y_{ijt}, Y_{ijt'}) = \theta_{j,T+1}$, que corresponde à estrutura gerada pela matriz $\mathbf{A}$ em (3).

Quando o comportamento médio longitudinal da variável resposta deve ser

descrito por uma função não linear no tempo, então

$$\mathbf{E}(Y_{ijt}) = \lambda_{jt} = \theta_{jt} + \theta_{j,T+1} = f_j(\text{tempo}_t, \boldsymbol{\beta}_j),$$

sendo $\boldsymbol{\beta}_j$ o vetor de parâmetros da função não linear correspondente ao tratamento j . Assim, obtém-se $f_j(\text{tempo}_t, \boldsymbol{\beta}_j) = \theta_{jt} + \theta_{j,T+1}$, e, por consequência:

$$\theta_{jt} = f_j(\text{tempo}_t, \boldsymbol{\beta}_j) - \theta_{j,T+1}. \quad (5)$$

Em (5), θ_{jt} é escrito em função do tempo t e do vetor de parâmetros da função não linear associado ao tratamento j , $\boldsymbol{\beta}_j$. Por esse motivo, a partir desse ponto ele será denotado por $\theta_{jt}(\boldsymbol{\beta}_j)$. Para o tratamento j tem-se $\boldsymbol{\theta}_j(\boldsymbol{\beta}_j) = (\theta_{j1}(\boldsymbol{\beta}_j), \theta_{j2}(\boldsymbol{\beta}_j), \dots, \theta_{jt}(\boldsymbol{\beta}_j))^t$.

A estimação dos parâmetros é feita por máxima verossimilhança, usando-se o algoritmo EM. No passo E, encontram-se as esperanças das variáveis latentes W_{ijt} condicionais aos valores observados $\mathbf{Y}_{ij}$ e às estimativas dos parâmetros da distribuição Poisson multivariada associada ao tratamento j após p iterações, ou seja, $s_{ijt} = \mathbf{E}(W_{ijt} \mid \mathbf{Y}_{ij}, \boldsymbol{\theta}_j(\boldsymbol{\beta}_j^{(p)}), \theta_{j,T+1}^{(p)})$. Mas:

$$\begin{aligned} Y_{ijt} &= W_{ijt} + W_{ij,T+1} \Rightarrow \\ s_{ijt} &= \mathbf{E}(W_{ijt} \mid \mathbf{Y}_{ij}, \boldsymbol{\theta}_j(\boldsymbol{\beta}_j^{(p)}), \theta_{j,T+1}^{(p)}) \\ &= \mathbf{E}(Y_{ijt} \mid \mathbf{Y}_{ij}, \boldsymbol{\theta}_j(\boldsymbol{\beta}_j^{(p)}), \theta_{j,T+1}^{(p)}) - \mathbf{E}(W_{ij,T+1} \mid \mathbf{Y}_{ij}, \boldsymbol{\theta}_j(\boldsymbol{\beta}_j^{(p)}), \theta_{j,T+1}^{(p)}) \\ &= Y_{ijt} - \mathbf{E}(W_{ij,T+1} \mid \mathbf{Y}_{ij}, \boldsymbol{\theta}_j(\boldsymbol{\beta}_j^{(p)}), \theta_{j,T+1}^{(p)}), \end{aligned} \quad (6)$$

qualquer que seja $i = 1, \dots, n_j$, $j = 1, \dots, k$ e $t = 1, \dots, T$. Dessa forma, é suficiente encontrar $\mathbf{E}(W_{ij,T+1} \mid \mathbf{Y}_{ij}, \boldsymbol{\theta}_j(\boldsymbol{\beta}_j^{(p)}), \theta_{j,T+1}^{(p)})$, e as demais são determinadas através da diferença em (6).

Tem-se, então:

$$\begin{aligned} s_{ij,T+1} &= \frac{\sum_{\mathbf{w}_{ij} \in g^{-1}(\mathbf{y}_{ij})} w_{ij,T+1} \mathbf{P}(W_{ij,T+1} = w_{ij,T+1}, \mathbf{Y}_{ij} = \mathbf{y}_{ij} \mid \boldsymbol{\theta}_j(\boldsymbol{\beta}_j^{(p)}), \theta_{j,T+1}^{(p)})}{\mathbf{P}(\mathbf{Y}_{ij} = \mathbf{y}_{ij} \mid \boldsymbol{\theta}_j(\boldsymbol{\beta}_j^{(p)}), \theta_{j,T+1}^{(p)})} \\ &= \frac{\sum_{w_{ij,T+1}=0}^{\min(y_{ij1}, \dots, y_{ijT})} w_{ij,T+1} \mathbf{P}(W_{ij,T+1} = w_{ij,T+1}, Y_{ij1} = y_{ij1}, \dots, Y_{ijT} = y_{ijT} \mid \boldsymbol{\theta}_j(\boldsymbol{\beta}_j^{(p)}), \theta_{j,T+1}^{(p)})}{\mathbf{P}(Y_{ij1} = y_{ij1}, \dots, Y_{ijT} = y_{ijT} \mid \boldsymbol{\theta}_j(\boldsymbol{\beta}_j^{(p)}), \theta_{j,T+1}^{(p)})} \\ &= \frac{\sum_{w_{ij,T+1}=0}^{\min(y_{ij1}, \dots, y_{ijT})} w_{ij,T+1} p_{ij}}{\sum_{w_{ij,T+1}=0}^{\min(y_{ij1}, \dots, y_{ijT})} p_{ij}}. \end{aligned}$$

em que

$$p_{ij} = \mathbf{P}(W_{ij,T+1} = w_{ij,T+1}, W_{ij1} = y_{ij1} - w_{ij,T+1}, \dots, W_{ijT} = y_{ijT} - w_{ij,T+1} \mid \boldsymbol{\theta}_j(\boldsymbol{\beta}_j^{(p)}), \theta_{j,T+1}^{(p)}).$$

Por independência, tem-se que:

$$p_{ij} = \frac{\exp(-\theta_{j,T+1}) \theta_{j,T+1}^{w_{ij,T+1}}}{(w_{ij,T+1})!} \prod_{t=1}^T \frac{\exp(-\theta_{jt}(\beta_j^{(p)})) \theta_{jt}(\beta_j^{(p)})^{y_{ijt}-w_{ij,T+1}}}{(y_{ijt}-w_{ij,T+1})!} \quad (7)$$

Os θ_{jt} , $t = 1, \dots, T$ em (7) são substituídos pelas suas respectivas expressões apresentadas em (5).

Os valores $s_{ij,T+1}$ obtidos no Passo E são tratados como verdadeiras realizações de $W_{ij,T+1}$, e utilizados no Passo M, no processo de maximização da verossimilhança.

Supondo que todas as observações das unidades experimentais são independentes, e identicamente distribuídas aquelas que receberam o mesmo tratamento, ou seja, $Y_{ijt} \sim P(\lambda_{jt})$, para todo $i = 1, \dots, n$, a função de verossimilhança conjunta é dada por:

$$\begin{aligned} L(\theta(\beta), \theta_{T+1} \mid \mathbf{Y}, \mathbf{W}_{j,T+1}) &= \prod_{j=1}^k P(\mathbf{Y}_j) = \prod_{j=1}^k \prod_{i=1}^{n_j} P(\mathbf{Y}_{ij}) \\ &= \prod_{j=1}^k \prod_{i=1}^{n_j} P(Y_{ij1} = y_{ij1}, Y_{ij2} = y_{ij2}, \dots, Y_{ijT} = y_{ijT}) \\ &= \prod_{j=1}^k \prod_{i=1}^{n_j} P(W_{ij,T+1} = s_{ij,T+1}, W_{ij1} = y_{ij1} - s_{ij,T+1}, \dots, W_{ijT} = y_{ijT} - s_{ij,T+1}) \\ &= \prod_{j=1}^k \prod_{i=1}^{n_j} \left[\frac{\exp(-\theta_{j,T+1}) \theta_{j,T+1}^{s_{ij,T+1}}}{(s_{ij,T+1})!} \prod_{t=1}^T \frac{\exp(-\theta_{jt}(\beta_j)) \theta_{jt}(\beta_j)^{y_{ijt}-s_{ij,T+1}}}{(y_{ijt}-s_{ij,T+1})!} \right], \quad (8) \end{aligned}$$

cujo logaritmo da função de verossimilhança é dado por:

$$\begin{aligned} l(\theta(\beta), \theta_{T+1} \mid \mathbf{Y}, \mathbf{W}_{j,T+1}) &= \sum_{j=1}^k \sum_{i=1}^{n_j} \log \left\{ \left[\frac{\exp(-\theta_{j,T+1}) \theta_{j,T+1}^{s_{ij,T+1}}}{(s_{ij,T+1})!} \prod_{t=1}^T \frac{\exp(-\theta_{jt}(\beta_j)) \theta_{jt}(\beta_j)^{y_{ijt}-s_{ij,T+1}}}{(y_{ijt}-s_{ij,T+1})!} \right] \right\} \\ &= \sum_{j=1}^k \sum_{i=1}^{n_j} \{ -\theta_{j,T+1} + s_{ij,T+1} \log(\theta_{j,T+1}) - \log[(s_{ij,T+1})!] + \\ &\quad \sum_{t=1}^T [-\theta_{jt}(\beta_j) + (y_{ijt} - s_{ij,T+1}) \log(\theta_{jt}(\beta_j)) - \log[(y_{ijt} - s_{ij,T+1})!]] \}. \quad (9) \end{aligned}$$

A obtenção dos vetores $\theta(\beta^{(p+1)}) = (\theta_1(\beta_1^{(p+1)}), \theta_2(\beta_2^{(p+1)}), \dots, \theta_k(\beta_k^{(p+1)}))^t$ e $\theta_{T+1}^{(p+1)} = (\theta_{1,T+1}^{(p+1)}, \theta_{2,T+1}^{(p+1)}, \dots, \theta_{k,T+1}^{(p+1)})^t$ que maximizam (9) foi efetuada através do algoritmo quasi-Newton BFGS, descrito em Nocedal e Wright (1999), implementado no pacote *stats*, função *optim* do programa estatístico R.

2.2.2.2 Extensão para delineamento em blocos casualizados com efeitos fixos de bloco

Seja Y_{ijt} a variável aleatória correspondente a uma observação feita no bloco i , $i = 1, \dots, b$, do tratamento j , $j = 1, \dots, k$, no tempo t , $t = 1, \dots, T$, resultante de um experimento no delineamento em blocos casualizados. Suponha que $\mathbf{Y}_{ij}$, vetor de observações do i -ésimo bloco do j -ésimo tratamento segue distribuição Poisson T -variada. Da mesma forma que para o delineamento completamente aleatorizado, Y_{ijt} é escrito como combinação linear de variáveis latentes, $Y_{ijt} = W_{ijt} + W_{ij,T+1}$. No entanto, há mudanças na sua distribuição, já que ela agora é afetada pelo efeito do bloco i . Assim, denotando por b_i o efeito do bloco i , considerado no modelo como aditivo, supondo que a covariância entre os tempos seja constante no j -ésimo tratamento, tem-se que:

$$\begin{aligned} \mathbf{E}(Y_{ijt}) &= \lambda_{ijt} = \theta_{ijt} + \theta_{j,T+1} = f_j(\text{tempo}_t, \boldsymbol{\beta}_j) + b_i, \text{ e} \\ \theta_{ijt} &= f_j(\text{tempo}_t, \boldsymbol{\beta}_j) + b_i - \theta_{j,T+1}. \end{aligned} \quad (10)$$

Por ser função do tempo_t , dos parâmetros da função não linear do tratamento j e do efeito de bloco b_i , θ_{ijt} será a partir de agora denotado por $\theta_{ijt}(\boldsymbol{\beta}_j, b_i)$.

Para a estimação dos parâmetros do modelo adota-se o mesmo método utilizado para delineamento completamente aleatorizado, devendo-se observar, no entanto, que as variáveis aleatórias em um mesmo tratamento e tempo não são mais identicamente distribuídas, devido ao efeito de blocos. Dessa forma, nas expressões de verossimilhança e logaritmo da verossimilhança, em (8) e (9), respectivamente, θ_{jt} deve ser substituído por θ_{ijt} que, por sua vez, deve ser substituído por sua respectiva expressão em (10).

2.2.2.3 Extensão para delineamento em blocos casualizados com efeitos aleatórios de bloco

Quando o efeito de blocos é aleatório, considera-se:

$$Y_{ijt} \mid b_i = W_{ijt} \mid b_i + W_{ij,T+1} \mid b_i, \quad (11)$$

com $(W_{ijt} \mid b_i) \sim P(\theta_{ijt})$, $t = 1, \dots, T$ e $(W_{ij,T+1} \mid b_i) \sim P(\theta_{j,T+1})$. Dessa maneira,

$$\mathbf{E}(Y_{ijt} \mid b_i) = \lambda_{ijt} = \theta_{ijt} + \theta_{j,T+1} = f_j(\text{tempo}_t, \boldsymbol{\beta}_j) + b_i, \text{ e}$$

θ_{ijt} é dado por (10).

Por ser $Y_{ijt} \geq 0$, adotou-se $b_i \sim \text{gama}(1/\sigma_b^2, 1/\sigma_b^2)$, de forma que $\mathbf{E}(b_i) = 1$ e $V(b_i) = \sigma_b^2$, para todo $i = 1, \dots, b$. Portanto,

$$\mathbf{E}(Y_{ijt}) = \mathbf{E}(\mathbf{E}(Y_{ijt} | b_i)) = f_j(\text{tempo}_t, \boldsymbol{\beta}_j) + 1$$

e a matriz de covariâncias entre observações de um mesmo bloco é dada a seguir (para detalhes, ver Anexo A).

$$\text{Cov}(\mathbf{Y}) = \begin{bmatrix} f_1(1) + \sigma_b^2 + 1 & \theta_{1,T+1} + \sigma_b^2 & \dots & \theta_{1,T+1} + \sigma_b^2 & \dots & \sigma_b^2 & \sigma_b^2 & \dots & \sigma_b^2 \\ \theta_{1,T+1} + \sigma_b^2 & f_1(2) + \sigma_b^2 + 1 & \dots & \theta_{1,T+1} + \sigma_b^2 & \dots & \sigma_b^2 & \sigma_b^2 & \dots & \sigma_b^2 \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \theta_{1,T+1} + \sigma_b^2 & \theta_{1,T+1} + \sigma_b^2 & \dots & f_1(T) + \sigma_b^2 + 1 & \dots & \sigma_b^2 & \sigma_b^2 & \dots & \sigma_b^2 \\ \sigma_b^2 & \sigma_b^2 & \dots & \sigma_b^2 & \dots & \sigma_b^2 & \sigma_b^2 & \dots & \sigma_b^2 \\ \sigma_b^2 & \sigma_b^2 & \dots & \sigma_b^2 & \dots & \sigma_b^2 & \sigma_b^2 & \dots & \sigma_b^2 \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \sigma_b^2 & \sigma_b^2 & \dots & \sigma_b^2 & \dots & \sigma_b^2 & \sigma_b^2 & \dots & \sigma_b^2 \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \sigma_b^2 & \sigma_b^2 & \dots & \sigma_b^2 & \dots & f_k(1) + \sigma_b^2 + 1 & \theta_{k,T+1} + \sigma_b^2 & \dots & \theta_{k,T+1} + \sigma_b^2 \\ \sigma_b^2 & \sigma_b^2 & \dots & \sigma_b^2 & \dots & \theta_{k,T+1} + \sigma_b^2 & f_k(2) + \sigma_b^2 + 1 & \dots & \theta_{k,T+1} + \sigma_b^2 \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \sigma_b^2 & \sigma_b^2 & \dots & \sigma_b^2 & \dots & \theta_{k,T+1} + \sigma_b^2 & \theta_{k,T+1} + \sigma_b^2 & \dots & f_k(T) + \sigma_b^2 + 1 \end{bmatrix}. \quad (12)$$

em que $f_j(t) = f_j(\text{tempo}_t, \boldsymbol{\beta}_j)$, $j = 1, \dots, k$ e $t = 1, \dots, T$. Note que não há mais independência marginal entre todas as observações de todas as unidades experimentais devido ao efeito aleatório de blocos. Entretanto, observações de blocos diferentes são marginalmente independentes.

No algoritmo EM, são definidos como dados não observados do bloco i os elementos do vetor $(\mathbf{W}_{i,T+1}, b_i)^t = (W_{i1,T+1}, W_{i2,T+1}, \dots, W_{ik,T+1}, b_i)^t$. A distribuição conjunta de $(\mathbf{W}_{i,T+1}, b_i)^t$ é dada por

$$\begin{aligned} f(\mathbf{W}_{i,T+1}, b_i) &= f(\mathbf{W}_{i,T+1} | b_i) f(b_i) \\ &= f(W_{i1,T+1}, W_{i2,T+1}, \dots, W_{ik,T+1} | b_i) f(b_i) \\ &= f(b_i) \prod_{j=1}^k f(W_{ij,T+1} | b_i) \\ &= \frac{1}{\Gamma(1/\sigma_b^2)} \left(\frac{1}{\sigma_b^2}\right)^{\frac{1}{\sigma_b^2}} b_i^{\frac{1}{\sigma_b^2}-1} e^{-\frac{b_i}{\sigma_b^2}} \prod_{j=1}^k \left[\frac{e^{-\theta_{j,T+1}} \theta_{j,T+1}^{w_{ij,T+1}}}{w_{ij,T+1}!} \right], \end{aligned}$$

o que resulta em

$$f(\mathbf{W}_{i,T+1}, b_i) = \frac{1}{\Gamma(1/\sigma_b^2)} \left(\frac{1}{\sigma_b^2}\right)^{\frac{1}{\sigma_b^2}} \prod_{j=1}^k \left[\frac{e^{-\theta_{j,T+1}}}{w_{ij,T+1}!} \right] \times \\ \times \exp \left[\left(\frac{1}{\sigma_b^2} - 1 \right) \log(b_i) - \frac{b_i}{\sigma_b^2} + \sum_{j=1}^k w_{ij,T+1} \log(\theta_{j,T+1}) \right]. \quad (13)$$

Por (13), vê-se que $w_{ij,T+1}$, $j = 1, 2, \dots, k$ e b_i são estatísticas suficientes. Então o algoritmo é executado da forma descrita a seguir. No passo E, detalhado no Anexo B, encontram-se primeiramente a esperança de b_i condicional ao vetor de variáveis latentes no bloco i e aos parâmetros envolvidos, isto é,

$$b_i^* = \mathbf{E}(b_i \mid \mathbf{Y}_i, \boldsymbol{\theta}_{it}(\boldsymbol{\beta}^{(p)}, b_i), \boldsymbol{\theta}_{T+1}^{(p)}, \sigma_b^{2(p)}) \\ = \left\{ \int_0^\infty \left[b_i \prod_{j=1}^k \left(\sum_{w_{ij,T+1} \in \omega} \left(\frac{e^{-\theta_{j,T+1}^{(p)}} \theta_{j,T+1}^{(p)w_{ij,T+1}}}{w_{ij,T+1}!} \prod_{t=1}^T \left(\frac{e^{-\theta_{ijt}(\boldsymbol{\beta}_j^{(p)}, b_i)} \theta_{ijt}(\boldsymbol{\beta}_j^{(p)}, b_i)^{y_{ijt}-w_{ij,T+1}}}{(y_{ijt}-w_{ij,T+1})!} \right) \right) \right] \right\} \times \\ \times \frac{1}{\Gamma(1/\sigma_b^{(p)^2})} \left(\frac{1}{\sigma_b^{(p)^2}} \right)^{1/\sigma_b^{(p)^2}} \frac{1}{b_i^{\frac{1}{\sigma_b^{(p)^2}-1}} e^{-\frac{b_i}{\sigma_b^{(p)^2}}} db_i} \Bigg] \Bigg/ \\ \left\{ \int_0^\infty \left[\prod_{j=1}^k \left(\sum_{w_{ij,T+1} \in \omega} \left(\frac{e^{-\theta_{j,T+1}^{(p)}} \theta_{j,T+1}^{(p)w_{ij,T+1}}}{w_{ij,T+1}!} \prod_{t=1}^T \left(\frac{e^{-\theta_{ijt}(\boldsymbol{\beta}_j^{(p)}, b_i)} \theta_{ijt}(\boldsymbol{\beta}_j^{(p)}, b_i)^{y_{ijt}-w_{ij,T+1}}}{(y_{ijt}-w_{ij,T+1})!} \right) \right) \right] \right\} \times \\ \times \frac{1}{\Gamma(1/\sigma_b^{(p)^2})} \left(\frac{1}{\sigma_b^{(p)^2}} \right)^{1/\sigma_b^{(p)^2}} \frac{1}{b_i^{\frac{1}{\sigma_b^{(p)^2}-1}} e^{-\frac{b_i}{\sigma_b^{(p)^2}}} db_i} \Bigg] \Bigg\}, \quad (14)$$

em que $\boldsymbol{\beta}^{(p)} = (\boldsymbol{\beta}_1^{(p)}, \boldsymbol{\beta}_2^{(p)}, \dots, \boldsymbol{\beta}_k^{(p)})^t$ e $\omega = \{0, 1, \dots, \min(y_{ij1}, y_{ij2}, y_{ijT})\}$. Em seguida, encontra-se a esperança de $W_{ij,T+1}$ condicional a $\mathbf{Y}_i$, b_i^* , $\boldsymbol{\beta}_j^{(p)}$ e $\boldsymbol{\theta}_{j,T+1}^{(p)}$, dada por

$$w_{ij,T+1}^* = \mathbf{E}(W_{ij,T+1} \mid \mathbf{Y}_i, b_i = b_i^*, \boldsymbol{\beta}_j^{(p)}, \boldsymbol{\theta}_{j,T+1}^{(p)}) = \\ \frac{\sum_{w_{ij,T+1}=0}^{\min(y_{ij1}, \dots, y_{ijT})} \left[w_{ij,T+1} \frac{e^{-\theta_{j,T+1}^{(p)}} \theta_{j,T+1}^{(p)w_{ij,T+1}}}{w_{ij,T+1}!} \prod_{t=1}^T \left(\frac{e^{-\theta_{ijt}(\boldsymbol{\beta}_j^{(p)}, b_i^*)} \theta_{ijt}(\boldsymbol{\beta}_j^{(p)}, b_i^*)^{y_{ijt}-w_{ij,T+1}}}{(y_{ijt}-w_{ij,T+1})!} \right) \right]}{\sum_{w_{ij,T+1}=0}^{\min(y_{ij1}, \dots, y_{ijT})} \left[\frac{e^{-\theta_{j,T+1}^{(p)}} \theta_{j,T+1}^{(p)w_{ij,T+1}}}{w_{ij,T+1}!} \prod_{t=1}^T \left(\frac{e^{-\theta_{ijt}(\boldsymbol{\beta}_j^{(p)}, b_i^*)} \theta_{ijt}(\boldsymbol{\beta}_j^{(p)}, b_i^*)^{y_{ijt}-w_{ij,T+1}}}{(y_{ijt}-w_{ij,T+1})!} \right) \right]},$$

em que $\theta_{ijt}(\beta_j^{(p)}, b_i^*)$ é dado pela expressão em (10), substituindo-se b_i por b_i^* .

No passo M, encontram-se as estimativas dos elementos do vetor de parâmetros que maximizam o logaritmo da função de verossimilhança, considerando-se $w_{ij,T+1}^*$ como verdadeira realização de $W_{ij,T+1}$, isto é,

$$\begin{aligned}
 l(\theta(\beta, \mathbf{b}^*), \theta_{T+1}, \sigma_b^2 \mid \mathbf{Y}, \mathbf{W}^*, \mathbf{b}^*) &= \sum_{i=1}^b \left\{ \log \left[\int_0^\infty \left(\prod_{j=1}^k \left[\left(\frac{e^{-\theta_{j,T+1}} \theta_{j,T+1}^{w_{ij,T+1}^*}}}{w_{ij,T+1}^*!} \right) \times \right. \right. \right. \\
 &\quad \times \left. \left. \prod_{t=1}^T \left(\frac{e^{-\theta_{ijt}(\beta_j, b_i^*)} \theta_{ijt}(\beta_j, b_i^*)^{y_{ijt} - w_{ij,T+1}^*}}{(y_{ijt} - w_{ij,T+1}^*)!} \right) \right] \times \right. \\
 &\quad \times \left. \left. \frac{1}{\Gamma(1/\sigma_b^2)} \left(\frac{1}{\sigma_b^2} \right)^{1/\sigma_b^2} b_i^{\frac{1}{\sigma_b^2}-1} e^{-\frac{b_i}{\sigma_b^2}} db_i \right) \right] \right\} \quad (15)
 \end{aligned}$$

As integrais em (14) e (15) são resolvidas através de integração numérica, utilizando método de quadratura Gaussiana. Para a maximização, novamente usa-se o algoritmo quasi-Newton BFGS.

2.2.2.4 Modelagem da superdispersão via distribuição Poisson multivariada mista

A modelagem de superdispersão, para delineamentos completamente ao acaso ou em modelos com efeitos fixos de bloco, recorre à utilização da distribuição Poisson multivariada mista, generalizando o caso bivariado apresentado em Karlis e Xekalaki (2005). Seja Z_{ij} uma variável aleatória em que $Z_{ij} \sim \text{gama}(1/\sigma_z^2, 1/\sigma_z^2)$, com $\mathbf{E}(Z_{ij}) = 1$ e $\text{V}(Z_{ij}) = \sigma_z^2$, $i = 1, \dots, n_j$ e $j = 1, \dots, k$. Seja também

$$Y_{ijt} \mid Z_{ij} = W_{ijt} \mid Z_{ij} + W_{ij,T+1} \mid Z_{ij}, \quad (16)$$

em que $(W_{ijt} \mid Z_{ij}) \sim P(\theta_{ijt}^*)$ e $(W_{ij,T+1} \mid Z_{ij}) \sim P(\theta_{ij,T+1}^*)$, com $\theta_{ijt}^* = \theta_{ijt} Z_{ij}$ e $\theta_{ij,T+1}^* = \theta_{j,T+1} Z_{ij}$, sendo W_{ijm} , $m = 1, \dots, T+1$, independentes. Por essa formulação, $(Y_{ijt} \mid Z_{ij}) \sim P(\lambda_{ijt})$, em que $\lambda_{ijt} = (\theta_{ijt} + \theta_{j,T+1}) Z_{ij}$. Supõe-se que:

$$\lambda_{ijt} = f_j(\text{tempo}_t, \beta_j) Z_{ij},$$

ou

$$\lambda_{ijt} = (f_j(\text{tempo}_t, \boldsymbol{\beta}_j) + b_i)Z_{ij},$$

para modelos com efeitos fixos de blocos. Chamando de $\mu_{ijt} = f_j(\text{tempo}_t, \boldsymbol{\beta}_j)$ ou, se houver efeitos fixos de blocos, $\mu_{ijt} = f_j(\text{tempo}_t, \boldsymbol{\beta}_j) + b_i$, tem-se que $(\theta_{ijt} + \theta_{j,T+1}) = \mu_{ijt}$. Então

$$\mathbf{E}(Y_{ijt}) = \mathbf{E}(\mathbf{E}(Y_{ijt} \mid Z_{ij})) = \mathbf{E}(\mu_{ijt}Z_{ij}) = \mu_{ijt}, \quad (17)$$

e

$$\begin{aligned} V(Y_{ijt}) &= \mathbf{E}(V(Y_{ijt} \mid Z_{ij})) + V(\mathbf{E}(Y_{ijt} \mid Z_{ij})) = \mathbf{E}(\mu_{ijt}Z_{ij}) + V(\mu_{ijt}Z_{ij}) \\ &= \mu_{ijt} + \mu_{ijt}^2 \sigma_z^2 = \mu_{ijt}(1 + \sigma_z^2 \mu_{ijt}). \end{aligned} \quad (18)$$

Por (17) e (18), vê-se que, $V(Y_{ijt}) > \mathbf{E}(Y_{ijt})$, permitindo a modelagem de superdispersão.

A covariância marginal entre medidas associada a uma mesma unidade experimental nos diferentes tempos é:

$$\text{Cov}(Y_{ijt}, Y_{ijt'}) = \mu_{ijt} \mu_{ijt'} \sigma_z^2 + \theta_{j,T+1}.$$

As demais medidas são independentes.

Para a estimação dos parâmetros, via algoritmo EM, considera-se como dados não observados da i -ésima observação do j -ésimo tratamento os elementos do vetor $(\mathbf{W}_{ij}, Z_{ij})^t$. Considerando

$$\boldsymbol{\theta}_j^{*(p)} = \boldsymbol{\theta}_j(\boldsymbol{\beta}_j^{(p)}) \quad (19)$$

para delineamentos completamente aleatorizados ou

$$\boldsymbol{\theta}_j^{*(p)} = \boldsymbol{\theta}_j(\boldsymbol{\beta}_j^{(p)}, b_i^{(p)}) \quad (20)$$

para delineamentos com efeitos fixos de blocos, a distribuição conjunta $f(\mathbf{W}_{ij}, Z_{ij})$ é dada por:

$$\begin{aligned}
f(\mathbf{W}_{ij}, Z_{ij} \mid \boldsymbol{\theta}_j^*, \theta_{j,T+1}, \sigma_z^2) &= f(\mathbf{W}_{ij} \mid Z_{ij}, \boldsymbol{\theta}_j^*, \theta_{j,T+1}, \sigma_z^2) f(Z_{ij} \mid \boldsymbol{\theta}_j^*, \theta_{j,T+1}, \sigma_z^2) \\
&= \frac{\exp(-\theta_{j,T+1} z_{ij}) (\theta_{j,T+1} z_{ij})^{w_{ij,T+1}}}{w_{ij,T+1}!} \prod_{t=1}^T \frac{\exp(-\theta_{jt}^* z_{ij}) (\theta_{jt}^* z_{ij})^{w_{ijt}}}{w_{ijt}!} \times \\
&\quad \times \frac{1}{\Gamma(1/\sigma_z^2)} \left(\frac{1}{\sigma_z^2}\right)^{1/\sigma_z^2} z_{ij}^{\frac{1}{\sigma_z^2}-1} \exp(-z_{ij}/\sigma_z^2) \\
&= \frac{1}{\Gamma(1/\sigma_z^2)} \left(\frac{1}{\sigma_z^2}\right)^{1/\sigma_z^2} \frac{1}{w_{ij,T+1}! \prod_{t=1}^T (w_{ijt}!)} \exp \left\{ -\theta_{j,T+1} z_{ij} - \sum_{t=1}^T \theta_{jt}^* z_{ij} \right. \\
&\quad + w_{ij,T+1} [\log(\theta_{j,T+1}) + \log(z_{ij})] + \sum_{t=1}^T [w_{ijt} (\log(\theta_{jt}^*) + \log(z_{ij}))] \\
&\quad \left. + \left(\frac{1}{\sigma_z^2} - 1\right) \log(z_{ij}) - \frac{z_{ij}}{\sigma_z^2} \right\}. \tag{21}
\end{aligned}$$

Por (21), vê-se que Z_{ij} , W_{ijt} , para todo $t = 1, \dots, T$ e $W_{ij,T+1}$ são estatísticas suficientes. Portanto, o Passo E do algoritmo EM consiste em primeiramente encontrar a esperança

$$\begin{aligned}
z_{ij}^* &= \mathbf{E}(Z_{ij} \mid \mathbf{Y}_{ij}, \boldsymbol{\theta}_j^{*(p)}, \theta_{j,T+1}^{(p)}, \sigma_z^{2(p)}) \\
&= \left\{ \sum_{w_{ij,T+1}=0}^{\min(y_{ij1}, \dots, y_{ijT})} \left[\frac{\theta_{j,T+1}^{(p)w_{ij,T+1}}}{w_{ij,T+1}!} \prod_{t=1}^T \left(\frac{\theta_{ijt}^{*(p)y_{ijt}-w_{ij,T+1}}}{(y_{ijt}-w_{ij,T+1})!} \right) \times \right. \right. \\
&\quad \times \left. \frac{\Gamma(w_{ij,T+1} + \sum_{t=1}^T (y_{ijt}-w_{ij,T+1}) + 1/\sigma_z^{(p)2} + 1)}{(\theta_{j,T+1} + \sum_{t=1}^T (\theta_{ijt}^{*(p)}) + 1/\sigma_z^{(p)2})^{w_{ij,T+1} + \sum_{t=1}^T (y_{ijt}-w_{ij,T+1}) + 1/\sigma_z^{(p)2} + 1}} \right] \Bigg\} / \\
&\quad \left\{ \sum_{w_{ij,T+1}=0}^{\min(y_{ij1}, \dots, y_{ijT})} \left[\frac{\theta_{j,T+1}^{(p)w_{ij,T+1}}}{w_{ij,T+1}!} \prod_{t=1}^T \left(\frac{\theta_{ijt}^{*(p)y_{ijt}-w_{ij,T+1}}}{(y_{ijt}-w_{ij,T+1})!} \right) \times \right. \right. \\
&\quad \times \left. \frac{\Gamma(w_{ij,T+1} + \sum_{t=1}^T (y_{ijt}-w_{ij,T+1}) + 1/\sigma_z^{(p)2})}{(\theta_{j,T+1} + \sum_{t=1}^T (\theta_{ijt}^{*(p)}) + 1/\sigma_z^{(p)2})^{w_{ij,T+1} + \sum_{t=1}^T (y_{ijt}-w_{ij,T+1}) + 1/\sigma_z^{(p)2}}} \right] \Bigg\}.
\end{aligned}$$

Em seguida, considerando-se z_{ij}^* como valor verdadeiro de Z_{ij} , encontram-se as esperanças condicionais $w_{ijm}^* = \mathbf{E}(W_{ijm} \mid \mathbf{Y}_{ij}, Z_{ij} = z_{ij}^*, \boldsymbol{\theta}_j^{*(p)}, \theta_{j,T+1}^{(p)}, \sigma_z^{2(p)})$, $m = 1, \dots, T+1$. Por (16), vê-se que é suficiente encontrar $w_{ij,T+1}^* = \mathbf{E}(W_{ij,T+1} \mid \mathbf{Y}_{ij}, Z_{ij} = z_{ij}^*, \boldsymbol{\Theta}^{(p)*})$, para

que as outras esperanças sejam determinadas por diferença. Assim, $w_{ij,T+1}^*$ é dada por:

$$w_{ij,T+1}^* = \frac{\sum_{w_{ij,T+1}=0}^{\min(y_{ij1}, \dots, y_{ijT})} \left\{ w_{ij,T+1} \frac{e^{(-\theta_{j,T+1}^{(p)} z_{ij}^*)} (\theta_{j,T+1}^{(p)} z_{ij}^*)^{w_{ij,T+1}}}{w_{ij,T+1}!} \prod_{t=1}^T \left[\frac{e^{(-\theta_{ijt}^{*(p)} z_{ij}^*)} (\theta_{ijt}^{*(p)} z_{ij}^*)^{(y_{ijt}-w_{ij,T+1})}}{(y_{ijt}-w_{ij,T+1})!} \right] \right\}}{\sum_{w_{ij,T+1}=0}^{\min(y_{ij1}, \dots, y_{ijT})} \left\{ \frac{e^{(-\theta_{j,T+1}^{(p)} z_{ij}^*)} (\theta_{j,T+1}^{(p)} z_{ij}^*)^{w_{ij,T+1}}}{w_{ij,T+1}!} \prod_{t=1}^T \left[\frac{e^{(-\theta_{ijt}^{*(p)} z_{ij}^*)} (\theta_{ijt}^{*(p)} z_{ij}^*)^{(y_{ijt}-w_{ij,T+1})}}{(y_{ijt}-w_{ij,T+1})!} \right] \right\}}$$

No passo M, encontram-se os valores dos parâmetros que maximizam o logaritmo da função de verossimilhança adotando-se $\mathbf{w}_{ij}^*$ como verdadeira realização de $\mathbf{w}_{ij}$, isto é,

$$\begin{aligned} l(\boldsymbol{\theta}^*, \boldsymbol{\theta}_{T+1}, \sigma_z^2 \mid \mathbf{Y}, \mathbf{W}^*) &= \sum_{j=1}^k \sum_{i=1}^{n_j} \left\{ \log\left(\frac{1}{\sigma_z^2}\right) - \log\left[\Gamma\left(\frac{1}{\sigma_z^2}\right)\right] + w_{ij,T+1}^* \log(\theta_{j,T+1}) - \log(w_{ij,T+1}!) \right. \\ &+ \sum_{t=1}^T [(y_{ijt} - w_{ij,T+1}) * \log(\theta_{ijt}^*) - \log[(y_{ijt} - w_{ij,T+1})!]] \\ &+ \log\left(\Gamma(w_{ij,T+1} + \sum_{t=1}^T (y_{ijt} - w_{ij,T+1}) + 1/\sigma_z^2)\right) \\ &\left. - (w_{ij,T+1} + \sum_{t=1}^T (y_{ijt} - w_{ij,T+1}) + \frac{1}{\sigma_z^2}) \log(\theta_{j,T+1} + \sum_{t=1}^T (\theta_{ijt}^*) + \frac{1}{\sigma_z^2}) \right\}. \end{aligned}$$

Note que $\theta_{ijt} = \mu_{ijt} - \theta_{j,T+1}$. $\boldsymbol{\theta}_j^{*(p)}$ é definido por (19) ou (20), dependendo do delineamento experimental. Para maiores detalhes de cálculos dos valores esperados condicionais e verossimilhança ver Anexo C.

A metodologia aqui proposta foi avaliada através de simulação e aplicada ao conjunto de dados de número de folhas em bromélia, apresentado na Tabela 17. O processo de simulação está descrito a seguir.

2.2.2.5 Simulação

A fim de avaliar a metodologia deste trabalho, foram simulados 1.000 conjuntos de dados com 4 tratamentos, 5 unidades de observação por tratamento e 7 medidas em cada unidade experimental, representando as medidas longitudinais. O comportamento médio longitudinal foi definido por uma curva logística para cada tratamento.

Estipulou-se que $\mu_{ijt} = \mathbf{E}(Y_{ijt}) = f_j(\text{tempo}_t, \boldsymbol{\beta}_j)$ para o delineamento completamente aleatorizado, e $\mu_{ijt} = \mathbf{E}(Y_{ijt}) = f_j(\text{tempo}_t, \boldsymbol{\beta}_j) + b_i$ para o delineamento em blocos casualizados com efeitos fixos de blocos, b_i representando o efeito de blocos, sendo

$$f_j(\text{tempo}_t, \boldsymbol{\beta}_j) = \frac{\alpha_j}{1 + \exp(-(\delta_j + \gamma_j \text{tempo}_t))}, \quad (22)$$

$i = 1, \dots, 5, j = 1, \dots, 4, \beta_j = (\alpha_j, \delta_j, \gamma_j)^t$, tempo = (1, 46, 91, 136, 181, 226, 271), tempo_{*t*} representando o *t*-ésimo elemento do vetor tempo, com $t = 1, \dots, 7$. Cada Y_{ijt} é definido como sendo combinação linear de duas variáveis aleatórias independentes, seguindo distribuição Poisson, isto é, $Y_{ijt} = W_{ijt} + W_{ijs}$, em que $W_{ijt} \sim P(\theta_{ijt})$ e $W_{ijs} \sim P(\theta_{js})$. O processo de simulação segue o apresentado em Krummenauer (1998). Inicialmente, gerou-se, para cada tratamento, uma amostra aleatória de tamanho 5 de $\mathbf{W}_{js}$, com distribuição $P(\theta_{js})$, obtendo-se cada $\mathbf{W}_{ijs}$. Em seguida, gerou-se $\mathbf{W}_{ijt}$, com distribuição $P(\theta_{ijt}) = P(\mu_{ijt} - \theta_{js})$, $i = 1, \dots, 5$. Finalmente, obteve-se uma amostra aleatória para cada tratamento, somando-se os valores gerados $Y_{ijt} = W_{ijt} + W_{ijs}$. Cada vetor, $\mathbf{Y}_{ij}$, de observações *i* do tratamento *j* tem distribuição Poisson multivariada de dimensão 7, em que θ_{js} é o parâmetro de covariância entre os tempos.

Supondo-se distribuição gama($1/\sigma_b^2, 1/\sigma_b^2$) para os efeitos aleatórios de blocos, foram gerados os valores de b'_i s, associados aos cinco blocos, e a seguir foram geradas as amostras de tamanho 5 de $(\mathbf{W}_{js} | b_i)$, com distribuição $P(\theta_{js})$, e $(\mathbf{W}_{ijt} | b_i)$, com distribuição $P(\theta_{ijt}) = P(\mu_{ijt} - \theta_{js})$, $\mu_{ijt} = f_j(\text{tempo}_t, \beta_j) + b_i$. Os valores da amostra observada para cada tratamento foram obtidos pela soma dos valores gerados $(Y_{ijt} | b_i) = (W_{ijt} | b_i) + (W_{ijs} | b_i)$.

O procedimento para obtenção da amostra de dados com superdispersão é similar ao realizado para o delineamento com blocos aleatorizados. Após se gerarem os valores para $Z_{ij} \sim \text{gama}(1/\sigma_z^2, 1/\sigma_z^2)$, para todo $i = 1, \dots, 5, j = 1, \dots, 4$, obteve-se $(\mathbf{W}_{ijs} | Z_{ij})$, com distribuição $P(\theta_{ijs}) = P(\theta_{js}Z_{ij})$ e $(\mathbf{W}_{ijt} | Z_{ij})$, com distribuição $P(\theta_{ijt}) = P((\mu_{ijt} - \theta_{js})Z_{ij})$. Determinaram-se então, os valores de $(Y_{ijt} | Z_{ij}) = (W_{ijt} | Z_{ij}) + (W_{ijs} | Z_{ij})$, para cada tratamento. Esse processo foi adotado, considerando-se delineamentos completamente aleatorizados ou com efeitos fixos de blocos.

A Figura 2 mostra o comportamento da curva teórica para cada tratamento, considerando-se os valores adotados para os parâmetros.

Para cada conjunto de dados foram estimados os parâmetros do modelo (22), os parâmetros de covariância θ_{js} , para todo $j = 1, 2, 3, 4$, os efeitos de bloco, b_i , $i = 1, 2, \dots, 5$ para modelos de efeitos fixos de blocos, além de σ_b^2 para modelos com efeitos aleatórios de blocos e σ_z^2 , para modelos que consideraram superdispersão, seguindo-

se a metodologia proposta. A avaliação dos resultados obtidos foi feita considerando-se as estatísticas média, variância, desvio padrão e erro quadrático médio, além de histogramas.

Tabela 2 - Valores dos parâmetros utilizados no processo de simulação^(*)

Tratamento				
Parâmetro	1	2	3	4
α	359,916	391,870	374,896	334,128
δ	-3,769	-3,441	-3,536	-3,981
γ	0,022	0,021	0,022	0,026
θ_8	8,000	12,000	10,000	6,000
Efeitos de blocos				
b_1	b_2	b_3	b_4	b_5
0,00	0,46	1,85	2,93	5,06

^(*) $\sigma_b^2 = 79,224$ e $\sigma_z^2 = 0,10$

Figura 2 - Curva teórica (média de cinco blocos) utilizada na simulação em cada um dos tratamentos

2.2.2.6 Análise dos dados de número de folhas em bromélias

Embora a análise exploratória tenha mostrado a existência de subdispersão, a metodologia proposta foi aplicada ao conjunto de dados de número de folhas de bromélias, descrito em (2.2.1). Será utilizada a distribuição Poisson hexavariada para o vetor aleatório $\mathbf{Y}_{ij}$ de medidas longitudinais do i -ésimo bloco, $i = 1, 2, \dots, 4$ e j -ésimo tratamento, $j = 1, 2, \dots, 5$, sob a suposição tanto de efeitos fixo quanto aleatório para blocos.

Nota-se que o comportamento médio do número de folhas em bromélias, por tratamento, observado na Figura 1 tem aparência de uma sigmóide, adotando-se a curva logística para representá-lo, cuja função é dada pela equação (22), sendo $j = 1, 2, \dots, 5$ e $t = 1, 2, \dots, 6$. Os efeitos de blocos são denotados por b_i , $i = 1, 2, \dots, 4$.

Várias situações quanto a diferenças entre tratamentos foram consideradas:

- (1) Mesma curva logística para todos os tratamentos, ou seja, $\alpha_j = \alpha_1$, $\delta_j = \delta_1$ e $\gamma_j = \gamma_1$, para todo $j = 1, 2, \dots, 5$;
- (2) igualdade entre os tratamentos 1 e 5, e mesmo comportamento médio para os tratamentos 2, 3 e 4, isto é, $\alpha_1 = \alpha_5$, $\delta_1 = \delta_5$, $\gamma_1 = \gamma_5$, $\alpha_2 = \alpha_3 = \alpha_4$, $\delta_2 = \delta_3 = \delta_4$ e $\gamma_2 = \gamma_3 = \gamma_4$;
- (3) tratamentos 3 e 4 com mesma curva média, o mesmo acontecendo com os tratamentos 1 e 5, ou seja, $\delta_1 = \delta_5$ e $\gamma_1 = \gamma_5$, $\alpha_3 = \alpha_4$, $\delta_3 = \delta_4$ e $\gamma_3 = \gamma_4$ e tratamento 2 diferente dos demais;
- (4) igualdade apenas entre os tratamentos 3 e 4;
- (5) todos os tratamentos diferentes.

Em cada um desses casos, encontraram-se as estimativas de máxima verossimilhança dos parâmetros, utilizando-se o algoritmo EM conforme proposto em (2.2.2.2) e (2.2.2.3). A comparação entre modelos foi feita usando-se o Critério de Informação da Akaike (AIC), dado por:

$$AIC = -2\log(\boldsymbol{\Theta} \mid \mathbf{Y}) + 2p,$$

em que $\boldsymbol{\Theta}$ é o vetor de parâmetros a serem estimados e p é a dimensão de $\boldsymbol{\Theta}$. Quanto menor o valor de AIC, melhor o modelo.

2.3 Resultados

2.3.1 Simulação

2.3.1.1 Delineamento completamente casualizado

A Tabela 3 apresenta as médias e os desvios padrões das estimativas dos parâmetros e a Tabela 4, os valores das variâncias e erros quadráticos médios dessas estimativas para o delineamento completamente aleatorizado.

Tabela 3 - Médias das estimativas (desvios padrões) obtidos na simulação para o delineamento completamente aleatorizado

Parâmetro	Tratamento			
	1	2	3	4
α	360,798 (13,428)	393,483 (13,939)	375,174 (12,543)	334,409 (9,144)
δ	-3,774 (0,097)	-3,444 (0,087)	-3,539 (0,096)	-3,989 (0,113)
γ	0,022 (0,001)	0,021 (0,001)	0,022 (0,001)	0,026 (0,001)
θ_8	8,225 (1,013)	12,219 (1,571)	10,721 (1,314)	6,256 (0,781)

Tabela 4 - Variância e erro quadrático médio das estimativas obtidas na simulação para o delineamento completamente aleatorizado

Parâmetro	Tratamento							
	1		2		3		4	
	Variância	EQM	Variância	EQM	Variância	EQM	Variância	EQM
α	180,3240	181,1012	194,2996	196,9020	157,3259	157,4030	83,6146	83,6937
δ	0,0094	0,0094	0,0075	0,0075	0,0092	0,0092	0,0127	0,0128
γ	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001
θ_8	1,0258	1,0762	2,4694	2,5173	1,7269	2,2464	0,6093	0,6748

Observa-se que as médias das estimativas para os parâmetros da curva logística estão bem próximas do verdadeiro valor dos parâmetros, com pequenos desvios padrões e os valores do erro quadrático médio e variância sendo também próximos, indicando um pequeno

viés e uma boa eficiência no processo de estimação. Para o parâmetro de covariância θ_8 , as médias também estão próximas do valor teórico do parâmetro, com desvios padrões não muito grandes. Também há proximidade entre os valores de variância e erro quadrático médio.

Os histogramas, apresentados no Anexo D, mostram uma distribuição simétrica das estimativas dos parâmetros da curva logística em relação ao valor verdadeiro do parâmetro. As distribuições das estimativas dos parâmetros de covariância são levemente assimétricas à esquerda, mas com maior concentração de valores próximos ao valor verdadeiro do parâmetro.

2.3.1.2 Delineamento em blocos casualizados com efeitos fixos de bloco

Na Tabela 5 encontram-se as médias e desvios padrões das estimativas dos parâmetros e na Tabela 6 estão os valores das variâncias e erros quadráticos médios dessas estimativas para o modelo com efeitos fixos de blocos.

Tabela 5 - Médias das estimativas (desvios padrões) obtidos na simulação para o modelo com efeitos fixos de blocos*

Parâmetro	Tratamento			
	1	2	3	4
α	360,643 (14,455)	392,659 (14,787)	375,316 (12,968)	334,280 (10,043)
δ	-3,777 (0,120)	-3,452 (0,105)	-3,546 (0,112)	-3,992 (0,145)
γ	0,022 (0,001)	0,021 (0,001)	0,022 (0,001)	0,026 (0,001)
θ_8	7,091 (1,569)	11,057 (1,880)	9,623 (1,598)	5,017 (1,545)
Efeitos de blocos				
	b_2	b_3	b_4	b_5
	0,71 (2,17)	1,90 (2,25)	2,89 (2,28)	5,07 (2,32)

* b_1 fixo, igual a zero.

Tabela 6 - Variância e erro quadrático médio das estimativas obtidas na simulação para o modelo com efeitos fixos de blocos

Tratamento								
	1		2		3		4	
Parâmetro	Variância	EQM	Variância	EQM	Variância	EQM	Variância	EQM
α	208,9618	209,4904	218,6678	219,2906	168,1668	168,3433	100,8678	100,8909
δ	0,0145	0,0146	0,0110	0,0111	0,0125	0,0126	0,0212	0,0213
γ	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001
θ_8	2,4613	3,2868	3,5346	4,4232	2,5534	2,6958	2,3879	3,3543
Efeitos de blocos								
	b_2		b_3		b_4		b_5	
	Variância	EQM	Variância	EQM	Variância	EQM	Variância	EQM
	4,7201	4,7829	5,0797	5,0820	5,1967	5,1979	5,3960	5,3962

Vê-se que as médias das estimativas dos parâmetros da curva logística estão próximas dos valores verdadeiros com desvios padrões pequenos, embora levemente maiores do que no caso do delineamento inteiramente casualizado. Os valores de variância e erro quadrático médio estão bem próximos, indicando um pequeno viés. De forma semelhante, as estimativas dos efeitos de blocos têm médias próximas dos valores verdadeiros, embora com desvios padrões grandes. Apenas a média das estimativas do efeito do bloco 2 está um pouco afastada do seu verdadeiro valor. Também há pequena diferença entre variância e erro quadrático médio. Já as médias das estimativas de θ_8 não estão tão próximas dos valores dos parâmetros quanto no delineamento completamente aleatorizado, com valores de desvios padrões razoavelmente grandes e valores de variância e erro quadrático médio um pouco distantes.

Os histogramas, apresentados no Anexo D, mostram que a distribuição das estimativas dos parâmetros são simétricas com relação aos verdadeiros valores, exceto para as estimativas dos parâmetros de covariância, que são assimétricas à esquerda, mas com maiores frequências nas classes próximas aos valores dos parâmetros.

2.3.1.3 Delineamento em blocos casualizados com efeitos aleatórios de bloco

Os resultados das simulações considerando-se como aleatórios os efeitos de blocos estão nas Tabelas 7 e 8. Pode-se ver que as médias das estimativas dos parâmetros da curva logística estão próximas dos valores teóricos e com erros padrões pequenos. Os

parâmetros de covariância têm as médias de suas estimativas um pouco menores do que os seus verdadeiros valores e com erros padrões grandes, comportamento semelhante ao do modelo com efeitos fixos de blocos. A média das estimativas de σ_b^2 , suposto igual a 79,224, foi igual a 2,99 com desvio padrão de 16,50, o que mostra que esse processo levou a estimativas viesadas para o parâmetro de variância intra-bloco, que, neste caso, está sendo subestimado. Além disso, os valores da variância das estimativas de σ_b^2 (272,2782) e erro quadrático médio (6083,3646) mostram um vício bastante grande, enfatizando a ineficiência desse método para estimar esse parâmetro. Esse comportamento das estimativas de σ_b^2 é decorrente do pequeno tamanho de amostra aqui considerada.

Tabela 7 - Médias das estimativas (desvios padrões) obtidos na simulação para o modelo com efeitos aleatórios de blocos

Parâmetro	Tratamento			
	1	2	3	4
α	358,303 (13,180)	390,645 (12,986)	373,832 (12,347)	332,916 (9,387)
δ	-3,807 (0,109)	-3,474 (0,091)	-3,571 (0,098)	-4,034 (0,118)
γ	0,022 (0,001)	0,021 (0,001)	0,022 (0,001)	0,026 (0,001)
θ_8	7,180 (1,262)	11,065 (1,529)	9,589 (1,452)	5,241 (1,012)

Tabela 8 - Variância e erro quadrático médio das estimativas obtidas na simulação para o modelo com efeitos aleatórios de blocos

Parâmetro	Tratamento							
	1		2		3		4	
	Variância	EQM	Variância	EQM	Variância	EQM	Variância	EQM
α	173,7038	176,3053	168,6256	170,1269	152,4570	153,5891	88,1135	89,5818
δ	0,0118	0,0133	0,0083	0,0094	0,0097	0,0109	0,0139	0,0168
γ	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001
θ_8	1,5933	2,2659	2,3388	3,2132	2,1094	2,2787	1,0243	1,6003

Observando-se os histogramas apresentados no Anexo D, vê-se que o histograma relativo às estimativas das variâncias dos efeitos de bloco é assimétrico à direita, com quase todos os valores entre 0 e 50, distantes do valor do parâmetro. Os histogramas associados às estimativas dos demais parâmetros são quase simétricos, com maior concentração de dados próximos aos verdadeiros valores dos parâmetros.

2.3.1.4 Delineamento completamente casualizado com superdispersão

Supondo a presença de superdispersão e o delineamento completamente casualizado, a Tabela 9 apresenta as médias e os desvios padrões das estimativas dos parâmetros e a Tabela 10 traz os valores das variâncias e erros quadráticos médios dessas estimativas. Os resultados mostram que as médias das estimativas dos parâmetros estão próximas dos valores verdadeiros, sendo que os erros padrões das assíntotas da curva logística (α) são maiores se comparados com os valores dessa medida nos casos anteriores, comportamento esperado já que as variâncias das variáveis aleatórias são maiores, afetando a variabilidade das estimativas dos parâmetros. Apesar disso, os valores das variâncias estão próximos dos valores dos erros quadráticos médios, indicando haver um viés pequeno, ocorrendo o mesmo para δ e γ . Para o parâmetro σ_z^2 , a média das estimativas foi igual a 0,0797 com desvio padrão de 0,0278, variância igual a 0,0008 e erro quadrático médio de 0,0012, também indicando presença de viés.

Tabela 9 - Médias das estimativas (desvios padrões) obtidos na simulação para o delineamento completamente aleatorizado na presença de superdispersão

Parâmetro	Tratamento			
	1	2	3	4
α	360,911 (52,509)	393,576 (57,825)	375,751 (53,929)	334,439 (49,850)
δ	-3,771 (0,103)	-3,444 (0,088)	-3,542 (0,091)	-3,985 (0,117)
γ	0,022 (0,001)	0,021 (0,001)	0,022 (0,001)	0,026 (0,001)
θ_8	8,348 (1,539)	12,481 (2,199)	10,838 (1,860)	6,303 (1,234)

Tabela 10 - Variância e erro quadrático médio das estimativas obtidas na simulação para o delineamento completamente aleatorizado na presença de superdispersão

Parâmetro	Tratamento							
	1		2		3		4	
	Variância	EQM	Variância	EQM	Variância	EQM	Variância	EQM
α	2757,1385	2758,1300	3343,6875	3346,5969	2908,3918	2909,1238	2485,0439	2485,1404
δ	0,0106	0,0106	0,0077	0,0077	0,0083	0,0083	0,0138	0,0138
γ	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001
θ_8	2,3680	2,4892	4,8350	5,0668	3,4589	4,1604	1,5232	1,6148

Os histogramas das estimativas dos parâmetros da curva logística e dos parâmetros de covariância são quase simétricos e com valores concentrados próximos aos valores dos parâmetros. Já o histograma relativo à σ_z^2 é assimétrico à direita, com uma maior frequência nas classes até o seu verdadeiro valor.

2.3.1.5 Delineamento com efeitos fixos de blocos na presença de superdispersão

A Tabela 11 apresenta as médias e os desvios padrões das estimativas dos parâmetros e na Tabela 12 encontram-se os valores das variâncias e erros quadráticos médios dessas estimativas.

As médias das estimativas das assíntotas das curvas logísticas estão um pouco afastadas do valor teórico adotado, com um desvio padrão grande, e os valores de variância e erro quadrático médio são diferentes sendo indicativo de presença de vício de estimativa. Comportamento semelhante tiveram os parâmetros de covariância. Os demais parâmetros da curva logística e de efeitos fixos de blocos, com exceção do efeito do bloco 5, tiveram as médias de suas estimativas próximas dos valores dos parâmetros, e valores de variância e erro quadrático médio também próximos. A média das estimativas de σ_z^2 foi 0,0855 com desvio padrão igual a 0,0893, variância igual a 0,0080 e erro quadrático médio de 0,0082, indicando uma grande variabilidade e um pequeno vício de estimativa.

Tabela 11 - Médias das estimativas (erros padrões) obtidos na simulação para o modelo com efeitos fixos de blocos na presença de superdispersão

Parâmetro	Tratamento			
	1	2	3	4
α	363,126 (52,827)	394,061 (57,545)	377,311 (54,350)	337,255 (49,919)
δ	-3,743 (0,137)	-3,425 (0,111)	-3,521 (0,120)	-3,944 (0,157)
γ	0,022 (0,001)	0,021 (0,001)	0,022 (0,001)	0,026 (0,001)
θ_8	6,776 (2,555)	10,114 (3,652)	8,758 (3,204)	5,021 (2,031)
Efeitos de blocos				
	b_2	b_3	b_4	b_5
	0,66 (5,48)	1,65 (5,83)	2,29 (5,21)	3,88 (5,69)

Tabela 12 - Variância e erro quadrático médio das estimativas obtidas na simulação para o modelo com efeitos fixos de blocos na presença de superdispersão

Parâmetro	Tratamento							
	1		2		3		4	
	Variância	EQM	Variância	EQM	Variância	EQM	Variância	EQM
α	2790,7198	2801,0266	3311,3997	3316,2006	2953,9786	2959,8115	2491,9227	2501,7032
δ	0,0187	0,0194	0,0124	0,0127	0,0144	0,0146	0,0248	0,0261
γ	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001
θ_8	6,5287	8,0268	13,3362	16,8944	10,2706	11,8139	4,1265	5,0841
Efeitos de blocos								
	b_2		b_3		b_4		b_5	
	Variância	EQM	Variância	EQM	Variância	EQM	Variância	EQM
	29,9997	30,0418	34,0418	34,0833	27,1655	27,5700	32,4241	33,8106

2.3.2 Análise dos dados de número de folhas em bromélias

2.3.2.1 Modelo com efeitos fixos de blocos

Os valores obtidos de AIC para os diferentes modelos apresentados em (2.2.2.6) estão na Tabela 13. Vê-se que são muito semelhantes. Entretanto, o modelo (2) tem o menor valor de AIC, sendo o mais parcimonioso dentre aqueles que consideram alguma diferença

entre os tratamentos. As estimativas dos parâmetros do modelo (2) e os respectivos erros padrões estão na Tabela 14, enquanto que a Figura 3 apresenta as curvas médias ajustadas e os valores médios observados para cada tratamento. A curva média que representa o comportamento dos tratamentos 1 e 5 mostra evidência de superioridade desses tratamentos se comparada à curva média relativa aos tratamentos 2, 3 e 4, mostrando então que o Xaxim (Tratamento 5) pode ser substituído por Casca de *Pinus* (Tratamento 1).

Tabela 13 - Valores de AIC para cada uma das suposições de igualdades entre parâmetros da curva logística - Efeitos fixos de blocos

Modelo	(1)	(2)	(3)	(4)	(5)
AIC	872,1980	863,2217	869,9676	875,7797	881,4938

Tabela 14 - Estimativas dos parâmetros e erros padrões do modelo (2) - Efeitos fixos de blocos

Parâmetro	Tratamento				
	1	2	3	4	5
α	222,118 (43,926)	304,844 (119,157)	304,844 (119,157)	304,844 (119,157)	222,118 (43,926)
δ	-1,248 (0,232)	-1,690 (0,439)	-1,690 (0,439)	-1,690 (0,439)	-1,248 (0,232)
γ	0,004 (0,001)	0,003 (0,001)	0,003 (0,001)	0,003 (0,001)	0,004 (0,001)
θ_7	4,304 (0,992)	0,031 (0,095)	1,067 (0,482)	1,613 (0,604)	12,437 (1,480)
Efeitos de blocos					
b_2	b_3		b_4		
0,918 (2,319)	-2,361 (2,281)		3,663 (2,337)

Figura 3 - Funções ajustadas e médias de quatro blocos observadas para os cinco tratamentos em cada tempo - Efeitos fixos de blocos

2.3.2.2 Modelo com efeitos aleatórios de blocos

Os resultados obtidos quando os efeitos de blocos foram considerados aleatórios (Tabelas 15 e 16) foram semelhantes aos de blocos fixos, sendo o modelo (2) o que apresentou menor AIC. No entanto, há uma menor diferença entre as estimativas das assíntotas das duas curvas, o que pode ser verificado na Figura 4. Os erros padrões das estimativas dos parâmetros das curvas foram menores para os Tratamentos 1 e 5 do que no caso de efeito fixo para bloco, ocorrendo o oposto para os demais tratamentos. As estimativas dos parâmetros de covariância e seus respectivos erros padrões foram maiores.

Tabela 15 - Valores de AIC para cada uma das suposições de igualdades entre parâmetros da curva logística - Efeitos aleatórios de blocos

Modelo	(1)	(2)	(3)	(4)	(5)
AIC	876,4194	865,5734	869,8566	874,8195	883,3104

Tabela 16 - Estimativas dos parâmetros do modelo (2) - Efeitos aleatórios de blocos⁽¹⁾

Parâmetro	Tratamento				
	1	2	3	4	5
α	231,083 (38,918)	273,630 (146,231)	273,630 (146,231)	273,630 (146,231)	231,083 (38,918)
δ	-1,305 (0,205)	-1,585 (0,599)	-1,585 (0,599)	-1,585 (0,599)	-1,305 (0,205)
γ	0,004 (0,001)	0,003 (0,001)	0,003 (0,001)	0,003 (0,001)	0,004 (0,001)
θ_7	11,653 (1,494)	0,953 (0,510)	8,083 (1,166)	9,567 (1,254)	13,284 (1,547)

⁽¹⁾ $\hat{\sigma}_b^2 = 0,0123(0,0249)$

Figura 4 - Funções ajustadas e médias de quatro blocos observadas para os cinco tratamentos em cada tempo - Efeitos aleatórios de blocos

Segundo a matriz apresentada em (12), a estrutura de covariância obtida para medidas de mesmo bloco e mesmo tratamento foi:

$$(i) \quad V(\mathbf{Y}_1) = V(\mathbf{Y}_5) = (52,09 \ 83,35 \ 95,55 \ 108,29 \ 121,26 \ 142,65)^t, \quad \text{Cov}(Y_{1t}, Y_{1t'}) = 11,67 \text{ e} \\ \text{Cov}(Y_{i5t}, Y_{i5t'}) = 13,30;$$

$$(ii) \quad V(\mathbf{Y}_2) = V(\mathbf{Y}_3) = V(\mathbf{Y}_4) = (49,22 \ 74,59 \ 85,20 \ 96,60 \ 108,68 \ 129,96)^t, \quad \text{Cov}(Y_{i2t}, Y_{i2t'}) = 0,97,$$

$$\text{Cov}(Y_{i3t}, Y_{i3t'}) = 8,10 \text{ e } \text{Cov}(Y_{i4t}, Y_{i4t'}) = 9,58,$$

em que $\mathbf{Y}_j = (\mathbf{Y}_{1j}, \mathbf{Y}_{2j}, \mathbf{Y}_{3j}, \mathbf{Y}_{4j})^t$ e $\mathbf{Y}_{ij} = (Y_{ij1}, Y_{ij2}, \dots, Y_{ij6})^t$. Para observações de um mesmo bloco e tratamentos diferentes, $\text{Cov}(Y_{ijt}, Y_{ij't'}) = 0,01$. As covariâncias entre as demais medidas são iguais a zero.

Vê-se que esses resultados são bastante diferentes dos constantes nas Tabelas 18 a 22, em que se encontram as estruturas de covariâncias observadas para todos os tratamentos. Observam-se covariâncias bastante distintas entre os tempos e ocorrem valores negativos para as mesmas. Essa diferença deve-se ao fato de que a metodologia aqui utilizada assume covariâncias iguais entre as medidas realizadas ao longo do tempo, e, pela própria característica da distribuição Poisson multivariada, apenas correlações positivas são possíveis. Portanto, conclui-se que uma modelagem que inclua essas características seria mais indicada.

3 CONSIDERAÇÕES FINAIS

Este trabalho abordou modelos não lineares para dados de contagem longitudinais através da distribuição Poisson multivariada. Foram considerados experimentos em delineamentos completamente casualizados e em blocos casualizados, com efeitos fixos e aleatórios de blocos, além da ocorrência de superdispersão. A estimação dos parâmetros foi realizada através do método de máxima verossimilhança via algoritmo EM. A avaliação da metodologia foi realizada através de simulação e foi aplicada a um conjunto de dados de número de folhas de bromélias, observadas ao longo do tempo.

Na maioria dos casos, as simulações indicaram eficiência do método na estimação dos parâmetros para o modelo, considerando o delineamento completamente casualizado, inclusive na ocorrência de superdispersão, e delineamento em blocos casualizados com efeito fixo (sem superdispersão) e aleatório para blocos. Entretanto, para o modelo com efeitos fixos de blocos na presença de superdispersão há indicação de existência de viés, sobretudo nas assíntotas do modelo não linear e nos parâmetros de covariância. O processo também não foi eficiente na estimação da variância dos efeitos aleatórios de blocos.

A metodologia proposta requer que as covariâncias entre os tempos sejam iguais, o que mostra a necessidade de estender os estudos de modo a permitir a utilização de estruturas de covariâncias menos restritivas. Uma outra restrição é não permitir modelar covariâncias negativas e subdispersão. Misturas finitas de distribuições Poisson multivariada foram abordadas por Karlis e Meligkotsidou (2006) e permitem a modelagem de superdispersão e covariâncias negativas e podem ser consideradas em uma extensão deste trabalho. A modelagem de subdispersão, no contexto aqui tratado, pode ser objeto de pesquisa.

A teoria de cópulas, que consiste em construir distribuições multivariadas pela especificação de marginais univariadas, pode também ser uma alternativa para modelar a estrutura de covariância gerada pelos dados de contagem longitudinais. Os estudos nesta área estão bastante desenvolvidos para dados contínuos. Pode-se citar o trabalho de Lindsey e Lindsey (2006) que apresentam ajustes de modelos não lineares para dados longitudinais, em que as estruturas de covariâncias foram geradas através da técnica de cópulas. Alguns trabalhos enfocando dados discretos já podem ser encontrados, como por exemplo, Nešlehová (2007), mas a literatura ainda é limitada, havendo, portanto, necessidade de mais

pesquisas nessa área.

REFERÊNCIAS

- ANDERSON, D.A.; AITKIN, M. Variance component models with binary response: interviewer variability. **Journal of the Royal Statistical Society**, Series B, London, v. 47, p. 203 – 210, 1985.
- ABRAMOWITZ, M. **Handbook of mathematical functions, with formulas, graphs, and mathematical tables**. New York: Dover Publications, 1972. 1046 p.
- BOX, G.E.P. Problems in the analysis of growth and wear curves. **Biometrics**, Washington, DC, v. 6, p. 362 – 389, 1950.
- BRESLOW, N.E.; CLAYTON, D.G. Approximate inference in generalized linear mixed models. **Journal of the American Statistical Association**, Boston, v. 88, n. 421, p. 9 – 25, 1993.
- COSTA, S.C. da **Modelos lineares generalizados mistos para dados longitudinais**, 2003. 110 p. Tese (Doutorado em Estatística e Experimentação Agronômica) - Escola Superior de Agricultura “Luiz de Queiroz”, Universidade de São Paulo, Piracicaba, 2003.
- COX, D.R. **The analysis of binary data**. London: Methuen, 1970. 142 p.
- DAVIDIAN, M.; GILTINAN, D.M. Some general estimation methods for nonlinear mixed effects models. **Journal of Biopharmaceutical Statistics**, New York, v. 3, p. 23 – 55, 1993.
- DAVIDIAN, M.; GILTINAN, D.M. **Nonlinear models for repeated measurement data**. London: Chapman & Hall, 1995. 359 p.
- DEMPSTER, A.P.; LAIRD, N.M.; RUBIN, D.B. Maximum likelihood from incomplete observations. **Journal of the Royal Statistical Society**, Series B, London, v. 39, p. 1 – 38, 1977.
- DIGGLE, P.J.; LIANG, K.Y.; ZEGER, S.L. **Analysis of longitudinal data**. New York: Oxford University Press, 2002. 253 p.
- ELSTON, R.C.; GRIZZLE, J.E. Time response curves and their confidence bands. **Biometrics**, Washington, DC, v. 18, p. 148 – 159, 1962.
- FEARN, T. A bayesian approach to the growth curves. **Biometrika**, Cambridge, v. 62, p. 89 – 100, 1975.
- GALLANT, A.R. Seemingly unrelated nonlinear regression. **Journal of Econometrics**, Amsterdam, v. 3, p. 35 – 50, 1975.
- GALLANT, A.R. **Nonlinear statistical models**. New York: John Wiley, 1987, 610 p.
- GEISSER, S.; GREENHOUSE, S.W. An extension of Box’s results on the use of F-distribution in multivariate analysis. **Annals of Mathematical Statistics**, Ann Arbor, v. 29, p. 885 – 891, 1958.

GENNINGS, C.; CHINCHILLI, V.M.; CARTER, W.H. Response surface analysis with correlated data: a nonlinear approach. **Journal of the American Statistical Association**, Boston, v. 84, p. 805 – 809, 1989.

GRIZZLE, J.E.; ALLEN, D.M. Analysis of dose and response curves. **Biometrics**, Washington, DC, v. 25, p. 357 – 382, 1969.

HARVILLE, T.A. Maximum likelihood approaches to variance component estimation and to related problems. **Journal of the American Statistical Association**, Boston, v. 72, p. 320 – 338, 1977.

HIRST, K.; ZERBE, G.O.; BOYLE, D.W.; WILKENING, R.B. On nonlinear random effects models for repeated measurements. **Communications in Statistics B: Simulation and Computation**, New York, v. 20, p. 463 – 478, 1991.

HO, L.L.; SINGER, J.M. Generalized least squares methods for bivariate Poisson regression. **Communications in Statistics - Theory and Methods**, New York, v. 30, p. 263 – 277, 2001.

HUI, S.L. Curve fitting for repeated measurements made at irregular time points. **Biometrics**, Washington, DC, v. 40, p. 691 – 697, 1984.

KARLIS, D.; MELIGKOTSIDOU, L. Multivariate Poisson regression with covariance structure. **Statistics and Computing**, London, v. 15, p. 255 – 265, 2005.

KARLIS, D.; MELIGKOTSIDOU, L. Finite mixtures of multivariate Poisson distributions with application. **Journal of Statistical Planning and Inference**, Amsterdam, Aug. 2006. In press.

KARLIS, D.; NTZOUFRAS, I. Analysis of sports data by using bivariate Poisson models. **The Statistician**, London, v. 52, p. 381 – 393, 2003.

KARLIS, D.; NTZOUFRAS, I. Bivariate Poisson and diagonal inflated bivariate Poisson regression models in R. **Journal of Statistical Software**, Los Angeles, v. 14, 2005. Disponível em: <<http://www.jstatsoft.org>>. Acesso em: 5 nov. 2005.

KARLIS, D.; XEKALAKI, E. Mixed Poisson distributions. **International Statistical Review**, Hague, v. 73, p. 35 – 58, 2005.

KOCH, G.C.; LANDIS, J.R.; FREEMAN, J.L.; FREEMAN, D.H.; LEHMAN, R.G. A general methodology for the analysis for repeated measurements of categorical data. **Biometrics**, Washington, DC, v. 38, p. 133 – 158, 1977.

KRUMMENAUER, F. Efficient simulation of multivariate binomial and Poisson distributions. **Biometrical Journal**, Berlin, v. 40, p. 823 – 832, 1998.

LAIRD, N.M.; WARE, J.H. Random-effects models for longitudinal data. **Biometrics**, Washington, DC, v. 38, p. 963 – 974, 1982.

LEE, Y.; NELDER, J.A. Hierarchical generalized linear model (with discussion). **Journal of the Royal Statistical Society**, London, Series B. v. 58, p. 619 – 678, 1996.

LI, C.-S.; LU, J.-C.; PARK, J.; KIM, K.; BRINKLEY, P.A.; PETERSON, J.P. Multivariate zero-inflated Poisson models and their applications. **Technometrics**, Alexandria, v. 41, n. 1, p. 29 – 38, 1999.

LIANG, K.Y.; ZEGER, S.L. Longitudinal data analysis using generalized linear models. **Biometrika**, Cambridge, v. 73, n. 1, p. 13 – 22, 1986.

LINDSEY, J.K.; LINDSEY, P.J. Multivariate distributions with correlation matrices for nonlinear repeated measurements. **Computational Statistics and Data Analysis**, Amsterdam, v. 50, p. 720 – 732, 2006.

LINDSTROM, M.J.; BATES, D.M. Nonlinear mixed effects models for repeated measures data. **Biometrics**, Washington, DC, v. 46, p. 673 – 687, 1990.

McCULLAGH, P. **Generalized linear models**. London: Chapman and Hall, 1983. 261 p.

McCULLAGH, P.; NELDER, J.A. Quasi-likelihood functions. **Annals of Statistics**, Hayward, v. 11, p. 59 – 67, 1983.

McCULLAGH, P.; NELDER, J.A. **Generalized linear models**. 2nd ed. London: Chapman and Hall, London, 1989. 511 p.

NELDER, J.A.; WEDDERBURN, R.W.M. Generalized linear models. **Journal of the Royal Statistical Society**, London, Series A, v. 135, p. 370 – 384, 1972.

NEŠLEHOVÁ, J. On rank correlation measures for non-continuous random variables. **Journal of Multivariate Analysis**, New York, v. 98, n. 3, Mar. 2007. In press.

NOCEDAL, J.; WRIGHT, S.J. **Numerical optimization**. New York: Springer-Verlag, 1999. 636p.

OCHI, Y.; PRENTICE, R.L. Likelihood inference incorrelated probit regression. **Biometrika**, Cambridge, v. 71, p. 531 – 543, 1984.

OGLIARI, P.J. **Modelos não lineares para dados longitudinais provenientes de experimentos em blocos casualizados**, 1998. 156 p. Tese (Doutorado em Estatística e Experimentação Agrônômica) - Escola Superior de Agricultura “Luiz de Queiroz”, Universidade de São Paulo, Piracicaba, 1998.

OGLIARI, P.J.; ANDRADE, D.F. Analysing longitudinal data via nonlinear models in randomized block designs. **Computational Statistics & Data Analysis**, Amsterdam, v. 36, p. 319 – 332, 2001.

POTHOFF, R.F.; ROY, S.N. A generalized multivariate analysis of variance model useful especially for growth curve problems. **Biometrika**, Cambridge, v. 51, p. 313 – 326, 1964.

RAO, C.R. Some problems involving linear hypotheses in multivariate analysis. **Biometrika**, Cambridge, v. 46, p. 49 – 58, 1959.

RAO, C.R. The theory of least squares when the parameters are stochastic and its applications to the analysis of growth curves. **Biometrika**, Cambridge, v. 52, p. 447 – 458, 1965.

RAO, C.R. Covariance adjustment and related problems in multivariate analysis. In: KRISHNAIAH, P.R. (Ed.). **Multivariate analysis II**. New York: Academic Press, 1966. p. 87 – 103.

RAO, C.R. Least squares theory using an estimated dispersion matrix and its application to measurement of signals. In: BERKELEY SYPOSIUM ON MATHEMATICAL, STATISTICS AND PROBABILITY, 5. , 1967, Berkeley. **Proceedings...** Berkeley: University of California Press, 1967. v. 1, p. 355 – 372.

R Development Core Team. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL <http://www.R-project.org/>, 2006

SCHALL, R. Estimation in generalized linear models with random effects. **Biometrika**, Cambridge, v. 78, p. 719 – 727, 1991.

SCHEINER, L.B.; BEAL, S.L. Evaluation of methods for estimating population pharmacokinetic parameters. I. Michaelis-Menten model: routine clinical pharmacokinetic data. **Journal of Pharmacokinetics and Biopharmaceutics**, San Francisco, v. 8, p. 553 – 571, 1980.

STIRATELLI, R.; LAIRD, N.; WARE, J.H. Random effects-models for serial observations with binary response. **Biometrics**, Washington, DC, v. 40, p. 961 – 971, 1984.

SINGER, J.M.; ANDRADE, D.F. Analysis of longitudinal data. In: SEN, P.K.; RAO, C.R.(Ed.). **Handbook of statistics**. Amsterdam: Elsevier, 2000. v. 18, p. 115 – 160.

TSIONAS, E. Bayesian multivariate Poisson regression. **Communications in Statistics - Theory and Methods**, New York, v. 30, p. 243 – 255, 2001.

VONESH, E.F. Nonlinear models for the analysis of longitudinal data. **Statistics in Medicine**, Chichester, v. 11, p. 1929 – 1954, 1992.

VONESH, E.F.; CARTER, R.L. Mixed-effects nonlinear regression for unbalanced repeated measures. **Biometrics**, Washington, DC, v. 48, p. 1 – 17, 1992.

VONESH, E.F.; CHINCHILLI, V.M. **Linear and nonlinear models for the analysis of repeated measurements**. New York: Marcel Dekker, 1997. 560 p.

VONESH, E.F.; WANG, H.; MAJUMDAR, D. Generalized least squares, Taylor series linearization, and Fisher's scoring in multivariate nonlinear regression. **Journal of the American Statistical Association**, Boston, v. 96, n. 453, p. 282 – 291, 2001.

VONESH, E.F.; WANG, H.; NIE, L.; MAJUMDAR, D. Conditional second-order generalized estimating equations for generalized linear and nonlinear mixed-effects models. **Journal of the American Statistical Association**, Boston, v. 97, n. 457, p. 271 – 283, 2002.

WACLAWIW, M.A.; LIANG, K.Y. Prediction of the random effects in the generalized linear model. **Journal of the American Statistical Association**, Boston, v. 88, p. 171 – 178, 1993.

WALHIN, J.F. Bivariate ZIP Models. **Biometrical Journal**, Berlin, v. 43. p. 147 – 160, 2001.

WANG, K.; LEE, A.H.; YAU, K.K.W.; CARRIVICK., P.J.W. A bivariate zero-inflated Poisson regression model to analyze occupational injuries. **Accident Analysis and Prevention**, Oxford, v. 35. p. 625 – 629, 2003.

WEDDERBURN, R.W.M. Quasi-likelihood functions, generalized linear models, and the Gauss-Newton method. **Biometrika**, Cambridge, v. 61, p. 439 – 447, 1974.

WISHART, J. Growth rate determination in nutrition studies with the bacon pig and their analysis. **Biometrika**, Cambridge, v. 30, p. 16 – 24, 1938.

XAVIER, L.H. **Modelos univariado e multivariado para análise de medidas repetidas e verificação da acurácia do modelo univariado por meio de simulação.** 2000. 91 p. Dissertação (Mestrado em Estatística e Experimentação Agronômica) - Escola Superior de Agricultura “Luiz de Queiroz”, Universidade de São Paulo, Piracicaba, 2000.

ZEGER, S.L.; LIANG, K.Y. Longitudinal data analysis for discrete and continuous outcomes. **Biometrics**, Washington, DC, v. 42, p. 121 – 130, 1986.

ZEGER, S.L.; LIANG, K.Y. An overview of methods for the analysis of longitudinal data. **Statistics in Medicine**, Chichester, v. 11, p. 1825 – 1839, 1992.

ZEGER, S.L.; LIANG, K.Y.; ALBERT, P. Models for longitudinal data: a generalized estimating equation approach. **Biometrics**, Washington, DC, v. 44, p. 1049 – 1060, 1988.

ZEGER, S.L.; LIANG, K.Y.; SELF, S.G. The analysis of binary longitudinal data with time-independent covariates. **Biometrika**, Cambridge, v. 72, p. 31 – 38, 1985.

APÊNDICES

APÊNDICE A - Conjunto de dados do experimento

Tabela 17 - Número total de folhas de bromélias por tratamento e tempo

Bloco	Tratamento	Tempo(dias após o plantio)					
		5	173	229	285	341	435
1	1	52	72	90	114	129	140
1	2	54	65	86	96	116	121
1	3	52	64	88	100	114	122
1	4	51	62	84	103	115	126
1	5	54	79	106	108	122	130
2	1	50	74	93	114	136	141
2	2	49	57	79	101	116	122
2	3	52	66	85	105	120	126
2	4	50	68	80	105	127	129
2	5	52	82	103	110	134	138
3	1	51	70	89	107	124	128
3	2	51	59	77	95	111	120
3	3	51	60	82	103	117	125
3	4	51	63	82	100	113	118
3	5	54	69	92	108	123	128
4	1	57	72	96	117	133	142
4	2	55	59	86	104	118	128
4	3	55	67	92	115	126	133
4	4	55	69	91	105	118	129
4	5	55	70	94	111	129	136

APÊNDICE B - Programas Computacionais utilizados na simulação

B.1 - Delineamento Completamente Aleatorizado

```
# PARÂMETROS DA CURVA LOGÍSTICA PARA O TRATAMENTO 1
alfa1=359.916; delta1=-3.769; gama1=0.022
# PARÂMETROS DA CURVA LOGÍSTICA PARA O TRATAMENTO 2
alfa2=391.870; delta2=-3.441; gama2=0.021
# PARÂMETROS DA CURVA LOGÍSTICA PARA O TRATAMENTO 3
alfa3=374.896; delta3=-3.536; gama3=0.022
# PARÂMETROS DA CURVA LOGÍSTICA PARA O TRATAMENTO 4
alfa4=334.128; delta4=-3.981; gama4=0.026

#Cálculos dos valores sobre a curva em cada tempo
t=c(1,46,91,136,181,226,271)

media = function(alfa,delta,gama,t){
  mu=matrix(numeric(length(t)),ncol=1)
  for (i in 1:length(t)){
    mu[i]=alfa/(1+exp(-(delta+gama*t[i])))
  }
  mu
}

mu1=media(alfa1,delta1,gama1,t); mu2=media(alfa2,delta2,gama2,t)
mu3=media(alfa3,delta3,gama3,t); mu4=media(alfa4,delta4,gama4,t)

#####Parâmetros de Covariância
theta18p=8; theta28p=12; theta38p=10; theta48p=6

n=5
trat=c(rep(1,n),rep(2,n),rep(3,n),rep(4,n))
trat1=c(rep(1,n),rep(0,3*n))
trat2=c(rep(0,n),rep(1,n),rep(0,2*n))
trat3=c(rep(0,2*n),rep(1,n),rep(0,n))
trat4=c(rep(0,3*n),rep(1,n))

#### Definição da função de verossimilhança a ser maximizada #####
#### Aqui, minimiza-se -log(verossimilhança) #####
#####(-1)*log(Função de verossimilhança)#####
fn <- function(p){
  lvero=(-(p[1]/(1+exp(-(p[2]+p[3]*t[1])))+p[1]/(1+exp(-(p[2]+p[3]*t[2])))+
    p[1]/(1+exp(-(p[2]+p[3]*t[3])))+p[1]/(1+exp(-(p[2]+p[3]*t[4])))+
    p[1]/(1+exp(-(p[2]+p[3]*t[5])))+p[1]/(1+exp(-(p[2]+p[3]*t[6])))+
    p[1]/(1+exp(-(p[2]+p[3]*t[7])))-
    6*p[4])+log(p[4])*s8+
```

$$\begin{aligned}
& (Yt1-s8)*\log(p[1]/(1+\exp(-(p[2]+p[3]*t[1]))))-p[4])+ \\
& (Yt2-s8)*\log(p[1]/(1+\exp(-(p[2]+p[3]*t[2]))))-p[4])+ \\
& (Yt3-s8)*\log(p[1]/(1+\exp(-(p[2]+p[3]*t[3]))))-p[4])+ \\
& (Yt4-s8)*\log(p[1]/(1+\exp(-(p[2]+p[3]*t[4]))))-p[4])+ \\
& (Yt5-s8)*\log(p[1]/(1+\exp(-(p[2]+p[3]*t[5]))))-p[4])+ \\
& (Yt6-s8)*\log(p[1]/(1+\exp(-(p[2]+p[3]*t[6]))))-p[4])+ \\
& (Yt7-s8)*\log(p[1]/(1+\exp(-(p[2]+p[3]*t[7]))))-p[4])- \\
& (\lgamma(s8+1)+\lgamma(Yt1-s8+1)+\lgamma(Yt2-s8+1)+ \\
& \lgamma(Yt3-s8+1)+\lgamma(Yt4-s8+1)+\lgamma(Yt5-s8+1)+ \\
& \lgamma(Yt6-s8+1)+\lgamma(Yt7-s8+1)))*\text{trat1}+ \\
& (- (p[5]/(1+\exp(-(p[6]+p[7]*t[1])))+p[5]/(1+\exp(-(p[6]+p[7]*t[2]))))+ \\
& p[5]/(1+\exp(-(p[6]+p[7]*t[3])))+p[5]/(1+\exp(-(p[6]+p[7]*t[4])))+ \\
& p[5]/(1+\exp(-(p[6]+p[7]*t[5])))+p[5]/(1+\exp(-(p[6]+p[7]*t[6])))+ \\
& p[5]/(1+\exp(-(p[6]+p[7]*t[7]))))- \\
& 6*p[8])+\log(p[8])*s8+ \\
& (Yt1-s8)*\log(p[5]/(1+\exp(-(p[6]+p[7]*t[1]))))-p[8])+ \\
& (Yt2-s8)*\log(p[5]/(1+\exp(-(p[6]+p[7]*t[2]))))-p[8])+ \\
& (Yt3-s8)*\log(p[5]/(1+\exp(-(p[6]+p[7]*t[3]))))-p[8])+ \\
& (Yt4-s8)*\log(p[5]/(1+\exp(-(p[6]+p[7]*t[4]))))-p[8])+ \\
& (Yt5-s8)*\log(p[5]/(1+\exp(-(p[6]+p[7]*t[5]))))-p[8])+ \\
& (Yt6-s8)*\log(p[5]/(1+\exp(-(p[6]+p[7]*t[6]))))-p[8])+ \\
& (Yt7-s8)*\log(p[5]/(1+\exp(-(p[6]+p[7]*t[7]))))-p[8])- \\
& (\lgamma(s8+1)+\lgamma(Yt1-s8+1)+\lgamma(Yt2-s8+1)+ \\
& \lgamma(Yt3-s8+1)+\lgamma(Yt4-s8+1)+\lgamma(Yt5-s8+1)+ \\
& \lgamma(Yt6-s8+1)+\lgamma(Yt7-s8+1)))*\text{trat2}+ \\
& (- (p[9]/(1+\exp(-(p[10]+p[11]*t[1])))+p[9]/(1+\exp(-(p[10]+p[11]*t[2]))))+ \\
& p[9]/(1+\exp(-(p[10]+p[11]*t[3])))+p[9]/(1+\exp(-(p[10]+p[11]*t[4])))+ \\
& p[9]/(1+\exp(-(p[10]+p[11]*t[5])))+p[9]/(1+\exp(-(p[10]+p[11]*t[6])))+ \\
& p[9]/(1+\exp(-(p[10]+p[11]*t[7]))))- \\
& 6*p[12])+\log(p[12])*s8+ \\
& (Yt1-s8)*\log(p[9]/(1+\exp(-(p[10]+p[11]*t[1]))))-p[12])+ \\
& (Yt2-s8)*\log(p[9]/(1+\exp(-(p[10]+p[11]*t[2]))))-p[12])+ \\
& (Yt3-s8)*\log(p[9]/(1+\exp(-(p[10]+p[11]*t[3]))))-p[12])+ \\
& (Yt4-s8)*\log(p[9]/(1+\exp(-(p[10]+p[11]*t[4]))))-p[12])+ \\
& (Yt5-s8)*\log(p[9]/(1+\exp(-(p[10]+p[11]*t[5]))))-p[12])+ \\
& (Yt6-s8)*\log(p[9]/(1+\exp(-(p[10]+p[11]*t[6]))))-p[12])+ \\
& (Yt7-s8)*\log(p[9]/(1+\exp(-(p[10]+p[11]*t[7]))))-p[12])- \\
& (\lgamma(s8+1)+\lgamma(Yt1-s8+1)+\lgamma(Yt2-s8+1)+ \\
& \lgamma(Yt3-s8+1)+\lgamma(Yt4-s8+1)+\lgamma(Yt5-s8+1)+ \\
& \lgamma(Yt6-s8+1)+\lgamma(Yt7-s8+1)))*\text{trat3}+ \\
& (- (p[13]/(1+\exp(-(p[14]+p[15]*t[1])))+p[13]/(1+\exp(-(p[14]+p[15]*t[2]))))+ \\
& p[13]/(1+\exp(-(p[14]+p[15]*t[3])))+p[13]/(1+\exp(-(p[14]+p[15]*t[4])))+ \\
& p[13]/(1+\exp(-(p[14]+p[15]*t[5])))+p[13]/(1+\exp(-(p[14]+p[15]*t[6])))+ \\
& p[13]/(1+\exp(-(p[14]+p[15]*t[7]))))-6*p[16])+\log(p[16])*s8+ \\
& (Yt1-s8)*\log(p[13]/(1+\exp(-(p[14]+p[15]*t[1]))))-p[16])+
\end{aligned}$$

```

(Yt2-s8)*log(p[13]/(1+exp(-(p[14]+p[15]*t[2])))-p[16])+
(Yt3-s8)*log(p[13]/(1+exp(-(p[14]+p[15]*t[3])))-p[16])+
(Yt4-s8)*log(p[13]/(1+exp(-(p[14]+p[15]*t[4])))-p[16])+
(Yt5-s8)*log(p[13]/(1+exp(-(p[14]+p[15]*t[5])))-p[16])+
(Yt6-s8)*log(p[13]/(1+exp(-(p[14]+p[15]*t[6])))-p[16])+
(Yt7-s8)*log(p[13]/(1+exp(-(p[14]+p[15]*t[7])))-p[16])-
(lgamma(s8+1)+lgamma(Yt1-s8+1)+lgamma(Yt2-s8+1)+
  lgamma(Yt3-s8+1)+lgamma(Yt4-s8+1)+lgamma(Yt5-s8+1)+
  lgamma(Yt6-s8+1)+lgamma(Yt7-s8+1)))*trat4
-sum(lvero)
}

##### Definição do gradiente da função a ser minimizada #####
grd <- deriv(
  ~(-(p1/(1+exp(-(p2+p3*t1)))+p1/(1+exp(-(p2+p3*t2)))+
    p1/(1+exp(-(p2+p3*t3)))+p1/(1+exp(-(p2+p3*t4)))+
    p1/(1+exp(-(p2+p3*t5)))+p1/(1+exp(-(p2+p3*t6)))+
    p1/(1+exp(-(p2+p3*t7)))-
    6*p4)+log(p4)*s8+
    (Yt1-s8)*log(p1/(1+exp(-(p2+p3*t1)))-p4)+
    (Yt2-s8)*log(p1/(1+exp(-(p2+p3*t2)))-p4)+
    (Yt3-s8)*log(p1/(1+exp(-(p2+p3*t3)))-p4)+
    (Yt4-s8)*log(p1/(1+exp(-(p2+p3*t4)))-p4)+
    (Yt5-s8)*log(p1/(1+exp(-(p2+p3*t5)))-p4)+
    (Yt6-s8)*log(p1/(1+exp(-(p2+p3*t6)))-p4)+
    (Yt7-s8)*log(p1/(1+exp(-(p2+p3*t7)))-p4)-
    (lgamma(s8+1)+lgamma(Yt1-s8+1)+lgamma(Yt2-s8+1)+
      lgamma(Yt3-s8+1)+lgamma(Yt4-s8+1)+lgamma(Yt5-s8+1)+
      lgamma(Yt6-s8+1)+lgamma(Yt7-s8+1)))*trat1+
    -(p5/(1+exp(-(p6+p7*t1)))+p5/(1+exp(-(p6+p7*t2)))+
      p5/(1+exp(-(p6+p7*t3)))+p5/(1+exp(-(p6+p7*t4)))+
      p5/(1+exp(-(p6+p7*t5)))+p5/(1+exp(-(p6+p7*t6)))+
      p5/(1+exp(-(p6+p7*t7)))-
      6*p8)+log(p8)*s8+
      (Yt1-s8)*log(p5/(1+exp(-(p6+p7*t1)))-p8)+
      (Yt2-s8)*log(p5/(1+exp(-(p6+p7*t2)))-p8)+
      (Yt3-s8)*log(p5/(1+exp(-(p6+p7*t3)))-p8)+
      (Yt4-s8)*log(p5/(1+exp(-(p6+p7*t4)))-p8)+
      (Yt5-s8)*log(p5/(1+exp(-(p6+p7*t5)))-p8)+
      (Yt6-s8)*log(p5/(1+exp(-(p6+p7*t6)))-p8)+
      (Yt7-s8)*log(p5/(1+exp(-(p6+p7*t7)))-p8)-
      (lgamma(s8+1)+lgamma(Yt1-s8+1)+lgamma(Yt2-s8+1)+
        lgamma(Yt3-s8+1)+lgamma(Yt4-s8+1)+lgamma(Yt5-s8+1)+
        lgamma(Yt6-s8+1)+lgamma(Yt7-s8+1)))*trat2+
      -(p9/(1+exp(-(p10+p11*t1)))+p9/(1+exp(-(p10+p11*t2)))+

```

```

p9/(1+exp(-(p10+p11*t3)))+p9/(1+exp(-(p10+p11*t4)))+
p9/(1+exp(-(p10+p11*t5)))+p9/(1+exp(-(p10+p11*t6)))+
p9/(1+exp(-(p10+p11*t7)))-
6*p12)+log(p12)*s8+
(Yt1-s8)*log(p9/(1+exp(-(p10+p11*t1)))-p12)+
(Yt2-s8)*log(p9/(1+exp(-(p10+p11*t2)))-p12)+
(Yt3-s8)*log(p9/(1+exp(-(p10+p11*t3)))-p12)+
(Yt4-s8)*log(p9/(1+exp(-(p10+p11*t4)))-p12)+
(Yt5-s8)*log(p9/(1+exp(-(p10+p11*t5)))-p12)+
(Yt6-s8)*log(p9/(1+exp(-(p10+p11*t6)))-p12)+
(Yt7-s8)*log(p9/(1+exp(-(p10+p11*t7)))-p12)-
(lgamma(s8+1)+lgamma(Yt1-s8+1)+lgamma(Yt2-s8+1)+
lgamma(Yt3-s8+1)+lgamma(Yt4-s8+1)+lgamma(Yt5-s8+1)+
lgamma(Yt6-s8+1)+lgamma(Yt7-s8+1)))*trat3+
(-(p13/(1+exp(-(p14+p15*t1)))+p13/(1+exp(-(p14+p15*t2)))+
p13/(1+exp(-(p14+p15*t3)))+p13/(1+exp(-(p14+p15*t4)))+
p13/(1+exp(-(p14+p15*t5)))+p13/(1+exp(-(p14+p15*t6)))+
p13/(1+exp(-(p14+p15*t7)))-
6*p16)+log(p16)*s8+
(Yt1-s8)*log(p13/(1+exp(-(p14+p15*t1)))-p16)+
(Yt2-s8)*log(p13/(1+exp(-(p14+p15*t2)))-p16)+
(Yt3-s8)*log(p13/(1+exp(-(p14+p15*t3)))-p16)+
(Yt4-s8)*log(p13/(1+exp(-(p14+p15*t4)))-p16)+
(Yt5-s8)*log(p13/(1+exp(-(p14+p15*t5)))-p16)+
(Yt6-s8)*log(p13/(1+exp(-(p14+p15*t6)))-p16)+
(Yt7-s8)*log(p13/(1+exp(-(p14+p15*t7)))-p16)-
(lgamma(s8+1)+lgamma(Yt1-s8+1)+lgamma(Yt2-s8+1)+
lgamma(Yt3-s8+1)+lgamma(Yt4-s8+1)+lgamma(Yt5-s8+1)+
lgamma(Yt6-s8+1)+lgamma(Yt7-s8+1)))*trat4),
c("p1", "p2", "p3", "p4", "p5", "p6", "p7", "p8", "p9", "p10", "p11", "p12", "p13",
  "p14", "p15", "p16"), function(p1, p2, p3, p4, p5, p6, p7, p8, p9, p10, p11, p12,
  p13, p14, p15, p16, t1, t2, t3, t4, t5, t6, t7) NULL)

sgrd <- function(p, Yt1, Yt2, Yt3, Yt4, Yt5, Yt6, Yt7, s8) {
  p1=p[1]; p2=p[2]; p3=p[3]; p4=p[4]; p5=p[5]; p6=p[6]; p7=p[7];
  p8=p[8]; p9=p[9]; p10=p[10]; p11=p[11]; p12=p[12]; p13=p[13];
  p14=p[14]; p15=p[15]; p16=p[16]
  t1=t[1]; t2=t[2]; t3=t[3]; t4=t[4]; t5=t[5]; t6=t[6]; t7=t[7]
  colSums(attr(grd(p1, p2, p3, p4, p5, p6, p7, p8, p9, p10, p11, p12, p13, p14, p15,
    p16, t1, t2, t3, t4, t5, t6, t7), "gradient"))
}

##### Início do Processo de Simulação #####

library(nlreg)

```



```

set.seed(33); smt=runif(1000)
smt=round(smt,digits=5); smt=smt*100000
result=matrix(0,nrow=2000,ncol=23)
k=1; nsimul=1
while (nsimul <= 1000){
  set.seed(smt[nsimul])

  #Variáveis latentes para o tratamento 1
  W11=rpois(n,mu1[1]-theta18p); W12=rpois(n,mu1[2]-theta18p)
  W13=rpois(n,mu1[3]-theta18p); W14=rpois(n,mu1[4]-theta18p)
  W15=rpois(n,mu1[5]-theta18p); W16=rpois(n,mu1[6]-theta18p)
  W17=rpois(n,mu1[7]-theta18p); W18=rpois(n,theta18p)

  #Variáveis latentes para o tratamento 2
  W21=rpois(n,mu2[1]-theta28p); W22=rpois(n,mu2[2]-theta28p)
  W23=rpois(n,mu2[3]-theta28p); W24=rpois(n,mu2[4]-theta28p)
  W25=rpois(n,mu2[5]-theta28p); W26=rpois(n,mu2[6]-theta28p)
  W27=rpois(n,mu2[7]-theta28p); W28=rpois(n,theta28p)

  #Variáveis latentes para o tratamento 3
  W31=rpois(n,mu3[1]-theta38p); W32=rpois(n,mu3[2]-theta38p)
  W33=rpois(n,mu3[3]-theta38p); W34=rpois(n,mu3[4]-theta38p)
  W35=rpois(n,mu3[5]-theta38p); W36=rpois(n,mu3[6]-theta38p)
  W37=rpois(n,mu3[7]-theta38p); W38=rpois(n,theta38p)

  #Variáveis latentes para o tratamento 4
  W41=rpois(n,mu4[1]-theta48p); W42=rpois(n,mu4[2]-theta48p)
  W43=rpois(n,mu4[3]-theta48p); W44=rpois(n,mu4[4]-theta48p)
  W45=rpois(n,mu4[5]-theta48p); W46=rpois(n,mu4[6]-theta48p)
  W47=rpois(n,mu4[7]-theta48p); W48=rpois(n,theta48p)

  #Variáveis observadas para o tratamento 1
  Y11=W11+W18; Y12=W12+W18; Y13=W13+W18; Y14=W14+W18
  Y15=W15+W18; Y16=W16+W18; Y17=W17+W18;

  Y1=cbind(Y11,Y12,Y13,Y14,Y15,Y16,Y17)
  Ytrat1=rbind(t(t(Y11)),t(t(Y12)),t(t(Y13)),t(t(Y14)),t(t(Y15)),t(t(Y16)),
               t(t(Y17)))

  #Variáveis observadas para o tratamento 2
  Y21=W21+W28; Y22=W22+W28; Y23=W23+W28; Y24=W24+W28;
  Y25=W25+W28; Y26=W26+W28; Y27=W27+W28;

  Y2=cbind(Y21,Y22,Y23,Y24,Y25,Y26,Y27)
  Ytrat2=rbind(t(t(Y21)),t(t(Y22)),t(t(Y23)),t(t(Y24)),t(t(Y25)),t(t(Y26)),

```

```

t(t(Y27)))

#Variáveis observadas para o tratamento 3
Y31=W31+W38; Y32=W32+W38; Y33=W33+W38; Y34=W34+W38
Y35=W35+W38; Y36=W36+W38; Y37=W37+W38

Y3=cbind(Y31,Y32,Y33,Y34,Y35,Y36,Y37)
Ytrat3=rbind(t(t(Y31)),t(t(Y32)),t(t(Y33)),t(t(Y34)),t(t(Y35)),t(t(Y36)),
             t(t(Y37)))

#Variáveis observadas para o tratamento 4
Y41=W41+W48; Y42=W42+W48; Y43=W43+W48; Y44=W44+W48
Y45=W45+W48; Y46=W46+W48; Y47=W47+W48;

Y4=cbind(Y41,Y42,Y43,Y44,Y45,Y46,Y47)
Ytrat4=rbind(t(t(Y41)),t(t(Y42)),t(t(Y43)),t(t(Y44)),t(t(Y45)),t(t(Y46)),
             t(t(Y47)))

Y=rbind(Y1,Y2,Y3,Y4)
Yt1=Y[,1]; Yt2=Y[,2]; Yt3=Y[,3]; Yt4=Y[,4]
Yt5=Y[,5]; Yt6=Y[,6]; Yt7=Y[,7];

tempo=c(rep(t[1],n),rep(t[2],n),rep(t[3],n),rep(t[4],n),rep(t[5],n),
        rep(t[6],n),rep(t[7],n))
Ytrat=data.frame(tempo,Ytrat1,Ytrat2,Ytrat3,Ytrat4)

#Encontrando valores iniciais para os parâmetros da regressão
init1 <- nlreg( Ytrat1 ~ (alfa1/(1+exp(-1*(delta1+gama1*tempo))))),
              start = c(alfa1=100, delta1=-1, gama1=0.05),
              data = Ytrat)
init2 <- nlreg( Ytrat2 ~ (alfa2/(1+exp(-1*(delta2+gama2*tempo))))),
              start = c(alfa2=100, delta2=-1, gama2=0.05),
              data = Ytrat)
init3 <- nlreg( Ytrat3 ~ (alfa3/(1+exp(-1*(delta3+gama3*tempo))))),
              start = c(alfa3=100, delta3=-1, gama3=0.05),
              data = Ytrat)
init4 <- nlreg( Ytrat4 ~ (alfa4/(1+exp(-1*(delta4+gama4*tempo))))),
              start = c(alfa4=100, delta4=-1, gama4=0.05),
              data = Ytrat)

alfa1=init1$coef[1]; delta1=init1$coef[2]; gama1=init1$coef[3]
alfa1_ini=alfa1; delta1_ini=delta1; gama1_ini=gama1
alfa2=init2$coef[1]; delta2=init2$coef[2]; gama2=init2$coef[3]
alfa2_ini=alfa2; delta2_ini=delta2; gama2_ini=gama2
alfa3=init3$coef[1]; delta3=init3$coef[2]; gama3=init3$coef[3]

```

```

alfa3_ini=alfa3; delta3_ini=delta3; gama3_ini=gama3
alfa4=init4$coef[1]; delta4=init4$coef[2]; gama4=init4$coef[3]
alfa4_ini=alfa4; delta4_ini=delta4; gama4_ini=gama4

#Encontrando os valores iniciais para os parâmetros de covariância
if (cov(Y11,Y12)>0) cov12_1=cov(Y11,Y12) else cov12_1=0
if (cov(Y11,Y13)>0) cov13_1=cov(Y11,Y13) else cov13_1=0
if (cov(Y11,Y14)>0) cov14_1=cov(Y11,Y14) else cov14_1=0
if (cov(Y11,Y15)>0) cov15_1=cov(Y11,Y15) else cov15_1=0
if (cov(Y11,Y16)>0) cov16_1=cov(Y11,Y16) else cov16_1=0
if (cov(Y11,Y17)>0) cov17_1=cov(Y11,Y17) else cov17_1=0
if (cov(Y12,Y13)>0) cov23_1=cov(Y12,Y13) else cov23_1=0
if (cov(Y12,Y14)>0) cov24_1=cov(Y12,Y14) else cov24_1=0
if (cov(Y12,Y15)>0) cov25_1=cov(Y12,Y15) else cov25_1=0
if (cov(Y12,Y16)>0) cov26_1=cov(Y12,Y16) else cov26_1=0
if (cov(Y12,Y17)>0) cov27_1=cov(Y12,Y17) else cov27_1=0
if (cov(Y13,Y14)>0) cov34_1=cov(Y13,Y14) else cov34_1=0
if (cov(Y13,Y15)>0) cov35_1=cov(Y13,Y15) else cov35_1=0
if (cov(Y13,Y16)>0) cov36_1=cov(Y13,Y16) else cov36_1=0
if (cov(Y13,Y17)>0) cov37_1=cov(Y13,Y17) else cov37_1=0
if (cov(Y14,Y15)>0) cov45_1=cov(Y14,Y15) else cov45_1=0
if (cov(Y14,Y16)>0) cov46_1=cov(Y14,Y16) else cov46_1=0
if (cov(Y14,Y17)>0) cov47_1=cov(Y14,Y17) else cov47_1=0
if (cov(Y15,Y16)>0) cov56_1=cov(Y15,Y16) else cov56_1=0
if (cov(Y15,Y17)>0) cov57_1=cov(Y15,Y17) else cov57_1=0
if (cov(Y16,Y17)>0) cov67_1=cov(Y16,Y17) else cov67_1=0

cov1=c(cov12_1,cov13_1,cov14_1,cov15_1,cov16_1,cov17_1,cov23_1,cov24_1,
      cov25_1,cov26_1,cov27_1,cov34_1,cov35_1,cov36_1,cov37_1,cov45_1,
      cov46_1,cov47_1,cov56_1,cov57_1,cov67_1)

theta18=min(mean(cov1),(alfa1/(1+exp(-1*(delta1+gama1*t[1]))))-0.00001,
            (alfa1/(1+exp(-1*(delta1+gama1*t[2]))))-0.00001,
            (alfa1/(1+exp(-1*(delta1+gama1*t[3]))))-0.00001,
            (alfa1/(1+exp(-1*(delta1+gama1*t[4]))))-0.00001,
            (alfa1/(1+exp(-1*(delta1+gama1*t[5]))))-0.00001,
            (alfa1/(1+exp(-1*(delta1+gama1*t[6]))))-0.00001,
            (alfa1/(1+exp(-1*(delta1+gama1*t[7]))))-0.00001)
theta18_ini=theta18

if (cov(Y21,Y22)>0) cov12_2=cov(Y21,Y22) else cov12_2=0
if (cov(Y21,Y23)>0) cov13_2=cov(Y21,Y23) else cov13_2=0
if (cov(Y21,Y24)>0) cov14_2=cov(Y21,Y24) else cov14_2=0
if (cov(Y21,Y25)>0) cov15_2=cov(Y21,Y25) else cov15_2=0
if (cov(Y21,Y26)>0) cov16_2=cov(Y21,Y26) else cov16_2=0

```

```

if (cov(Y21,Y27)>0) cov17_2=cov(Y21,Y27) else cov17_2=0
if (cov(Y22,Y23)>0) cov23_2=cov(Y22,Y23) else cov23_2=0
if (cov(Y22,Y24)>0) cov24_2=cov(Y22,Y24) else cov24_2=0
if (cov(Y22,Y25)>0) cov25_2=cov(Y22,Y25) else cov25_2=0
if (cov(Y22,Y26)>0) cov26_2=cov(Y22,Y26) else cov26_2=0
if (cov(Y22,Y27)>0) cov27_2=cov(Y22,Y27) else cov27_2=0
if (cov(Y23,Y24)>0) cov34_2=cov(Y23,Y24) else cov34_2=0
if (cov(Y23,Y25)>0) cov35_2=cov(Y23,Y25) else cov35_2=0
if (cov(Y23,Y26)>0) cov36_2=cov(Y23,Y26) else cov36_2=0
if (cov(Y23,Y27)>0) cov37_2=cov(Y23,Y27) else cov37_2=0
if (cov(Y24,Y25)>0) cov45_2=cov(Y24,Y25) else cov45_2=0
if (cov(Y24,Y26)>0) cov46_2=cov(Y24,Y26) else cov46_2=0
if (cov(Y24,Y27)>0) cov47_2=cov(Y24,Y27) else cov47_2=0
if (cov(Y25,Y26)>0) cov56_2=cov(Y25,Y26) else cov56_2=0
if (cov(Y25,Y27)>0) cov57_2=cov(Y25,Y27) else cov57_2=0
if (cov(Y26,Y27)>0) cov67_2=cov(Y26,Y27) else cov67_2=0

cov2=c(cov12_2,cov13_2,cov14_2,cov15_2,cov16_2,cov17_2,cov23_2,cov24_2,
      cov25_2,cov26_2,cov27_2,cov34_2,cov35_2,cov36_2,cov37_2,cov45_2,
      cov46_2,cov47_2,cov56_2,cov57_2,cov67_2)

theta28=min(mean(cov2),(alfa2/(1+exp(-1*(delta2+gama2*t[1]))))-0.00001,
            (alfa2/(1+exp(-1*(delta2+gama2*t[2]))))-0.00001,
            (alfa2/(1+exp(-1*(delta2+gama2*t[3]))))-0.00001,
            (alfa2/(1+exp(-1*(delta2+gama2*t[4]))))-0.00001,
            (alfa2/(1+exp(-1*(delta2+gama2*t[5]))))-0.00001,
            (alfa2/(1+exp(-1*(delta2+gama2*t[6]))))-0.00001,
            (alfa2/(1+exp(-1*(delta2+gama2*t[7]))))-0.00001)
theta28_ini=theta28

if (cov(Y31,Y32)>0) cov12_3=cov(Y31,Y32) else cov12_3=0
if (cov(Y31,Y33)>0) cov13_3=cov(Y31,Y33) else cov13_3=0
if (cov(Y31,Y34)>0) cov14_3=cov(Y31,Y34) else cov14_3=0
if (cov(Y31,Y35)>0) cov15_3=cov(Y31,Y35) else cov15_3=0
if (cov(Y31,Y36)>0) cov16_3=cov(Y31,Y36) else cov16_3=0
if (cov(Y31,Y37)>0) cov17_3=cov(Y31,Y37) else cov17_3=0
if (cov(Y32,Y33)>0) cov23_3=cov(Y32,Y33) else cov23_3=0
if (cov(Y32,Y34)>0) cov24_3=cov(Y32,Y34) else cov24_3=0
if (cov(Y32,Y35)>0) cov25_3=cov(Y32,Y35) else cov25_3=0
if (cov(Y32,Y36)>0) cov26_3=cov(Y32,Y36) else cov26_3=0
if (cov(Y32,Y37)>0) cov27_3=cov(Y32,Y37) else cov27_3=0
if (cov(Y33,Y34)>0) cov34_3=cov(Y33,Y34) else cov34_3=0
if (cov(Y33,Y35)>0) cov35_3=cov(Y33,Y35) else cov35_3=0
if (cov(Y33,Y36)>0) cov36_3=cov(Y33,Y36) else cov36_3=0
if (cov(Y33,Y37)>0) cov37_3=cov(Y33,Y37) else cov37_3=0

```

```

if (cov(Y34,Y35)>0) cov45_3=cov(Y34,Y35) else cov45_3=0
if (cov(Y34,Y36)>0) cov46_3=cov(Y34,Y36) else cov46_3=0
if (cov(Y34,Y37)>0) cov47_3=cov(Y34,Y37) else cov47_3=0
if (cov(Y35,Y36)>0) cov56_3=cov(Y35,Y36) else cov56_3=0
if (cov(Y35,Y37)>0) cov57_3=cov(Y35,Y37) else cov57_3=0
if (cov(Y36,Y37)>0) cov67_3=cov(Y36,Y37) else cov67_3=0

cov3=c(cov12_3,cov13_3,cov14_3,cov15_3,cov16_3,cov17_3,cov23_3,cov24_3,
      cov25_3,cov26_3,cov27_3,cov34_3,cov35_3,cov36_3,cov37_3,cov45_3,
      cov46_3,cov47_3,cov56_3,cov57_3,cov67_3)

theta38=min(mean(cov3),(alfa3/(1+exp(-1*(delta3+gama3*t[1]))))-0.00001,
            (alfa3/(1+exp(-1*(delta3+gama3*t[2]))))-0.00001,
            (alfa3/(1+exp(-1*(delta3+gama3*t[3]))))-0.00001,
            (alfa3/(1+exp(-1*(delta3+gama3*t[4]))))-0.00001,
            (alfa3/(1+exp(-1*(delta3+gama3*t[5]))))-0.00001,
            (alfa3/(1+exp(-1*(delta3+gama3*t[6]))))-0.00001,
            (alfa3/(1+exp(-1*(delta3+gama3*t[7]))))-0.00001)
theta38_ini=theta38

if (cov(Y41,Y42)>0) cov12_4=cov(Y41,Y42) else cov12_4=0
if (cov(Y41,Y43)>0) cov13_4=cov(Y41,Y43) else cov13_4=0
if (cov(Y41,Y44)>0) cov14_4=cov(Y41,Y44) else cov14_4=0
if (cov(Y41,Y45)>0) cov15_4=cov(Y41,Y45) else cov15_4=0
if (cov(Y41,Y46)>0) cov16_4=cov(Y41,Y46) else cov16_4=0
if (cov(Y41,Y47)>0) cov17_4=cov(Y41,Y47) else cov17_4=0
if (cov(Y42,Y43)>0) cov23_4=cov(Y42,Y43) else cov23_4=0
if (cov(Y42,Y44)>0) cov24_4=cov(Y42,Y44) else cov24_4=0
if (cov(Y42,Y45)>0) cov25_4=cov(Y42,Y45) else cov25_4=0
if (cov(Y42,Y46)>0) cov26_4=cov(Y42,Y46) else cov26_4=0
if (cov(Y42,Y47)>0) cov27_4=cov(Y42,Y47) else cov27_4=0
if (cov(Y43,Y44)>0) cov34_4=cov(Y43,Y44) else cov34_4=0
if (cov(Y43,Y45)>0) cov35_4=cov(Y43,Y45) else cov35_4=0
if (cov(Y43,Y46)>0) cov36_4=cov(Y43,Y46) else cov36_4=0
if (cov(Y43,Y47)>0) cov37_4=cov(Y43,Y47) else cov37_4=0
if (cov(Y44,Y45)>0) cov45_4=cov(Y44,Y45) else cov45_4=0
if (cov(Y44,Y46)>0) cov46_4=cov(Y44,Y46) else cov46_4=0
if (cov(Y44,Y47)>0) cov47_4=cov(Y44,Y47) else cov47_4=0
if (cov(Y45,Y46)>0) cov56_4=cov(Y45,Y46) else cov56_4=0
if (cov(Y45,Y47)>0) cov57_4=cov(Y45,Y47) else cov57_4=0
if (cov(Y46,Y47)>0) cov67_4=cov(Y46,Y47) else cov67_4=0

cov4=c(cov12_4,cov13_4,cov14_4,cov15_4,cov16_4,cov17_4,cov23_4,cov24_4,
      cov25_4,cov26_4,cov27_4,cov34_4,cov35_4,cov36_4,cov37_4,cov45_4,
      cov46_4,cov47_4,cov56_4,cov57_4,cov67_4)

```

```

theta48=min(mean(cov4),(alfa4/(1+exp(-1*(delta4+gama4*t[1]))))-0.00001,
             (alfa4/(1+exp(-1*(delta4+gama4*t[2]))))-0.00001,
             (alfa4/(1+exp(-1*(delta4+gama4*t[3]))))-0.00001,
             (alfa4/(1+exp(-1*(delta4+gama4*t[4]))))-0.00001,
             (alfa4/(1+exp(-1*(delta4+gama4*t[5]))))-0.00001,
             (alfa4/(1+exp(-1*(delta4+gama4*t[6]))))-0.00001,
             (alfa4/(1+exp(-1*(delta4+gama4*t[7]))))-0.00001)
theta48_ini=theta48

```

```
#####Início do Processo Iterativo
```

```

nk=length(Yt1)
m2lnL=1000; m2lnlike=1000
m2lnL1=0; m2lnL2=0
iter=0
difllike1=1000; difllike2=1000
prec=1e-6; maxit=500

```

```

while ((difllike1 > prec) & (iter < maxit )) {
iter=iter+1; m2lnL1=m2lnL; m2lnL2=m2lnlike

```

```
#####Iniciando o Passo E do Algoritmo EM#####
```

```

numer8=matrix((rep(0,nk)),nrow=nk,ncol=1);
denom=matrix((rep(0,nk)),nrow=nk,ncol=1)
s1=matrix((rep(0,nk)),nrow=nk,ncol=1); s2=matrix((rep(0,nk)),nrow=nk,ncol=1)
s3=matrix((rep(0,nk)),nrow=nk,ncol=1); s4=matrix((rep(0,nk)),nrow=nk,ncol=1)
s5=matrix((rep(0,nk)),nrow=nk,ncol=1); s6=matrix((rep(0,nk)),nrow=nk,ncol=1)
s7=matrix((rep(0,nk)),nrow=nk,ncol=1); s8=matrix((rep(0,nk)),nrow=nk,ncol=1)
lnlike=matrix((rep(0,nk)),nrow=nk,ncol=1)

```

```
#### Encontrando as esperanças das variáveis latentes #####
```

```

for (i in 1:nk){
  for (w8 in 0:min(Yt1[i],Yt2[i],Yt3[i],Yt4[i],Yt5[i],Yt6[i],Yt7[i]))){
    lp=(-1*theta18)+log(theta18**w8)+
      (-1*((alfa1/(1+exp(-1*(delta1+gama1*t[1]))))-theta18))+
      (-1*((alfa1/(1+exp(-1*(delta1+gama1*t[2]))))-theta18))+
      (-1*((alfa1/(1+exp(-1*(delta1+gama1*t[3]))))-theta18))+
      (-1*((alfa1/(1+exp(-1*(delta1+gama1*t[4]))))-theta18))+
      (-1*((alfa1/(1+exp(-1*(delta1+gama1*t[5]))))-theta18))+
      (-1*((alfa1/(1+exp(-1*(delta1+gama1*t[6]))))-theta18))+
      (-1*((alfa1/(1+exp(-1*(delta1+gama1*t[7]))))-theta18))+
      (Yt1[i]-w8)*log((((alfa1/(1+exp(-1*(delta1+gama1*t[1]))))-theta18)))+
      (Yt2[i]-w8)*log((((alfa1/(1+exp(-1*(delta1+gama1*t[2]))))-theta18)))+
      (Yt3[i]-w8)*log((((alfa1/(1+exp(-1*(delta1+gama1*t[3]))))-theta18)))+
      (Yt4[i]-w8)*log((((alfa1/(1+exp(-1*(delta1+gama1*t[4]))))-theta18)))+
      (Yt5[i]-w8)*log((((alfa1/(1+exp(-1*(delta1+gama1*t[5]))))-theta18)))+

```

```

(Yt6[i]-w8)*log((((alfa1/(1+exp(-1*(delta1+gama1*t[6]))))-theta18)))+
(Yt7[i]-w8)*log((((alfa1/(1+exp(-1*(delta1+gama1*t[7]))))-theta18)))-
lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-
lgamma(Yt3[i]-w8+1)-lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-
lgamma(Yt6[i]-w8+1)-lgamma(Yt7[i]-w8+1))*trat1[i]+
((-1*theta28)+log(theta28**w8)+
(-1*((alfa2/(1+exp(-1*(delta2+gama2*t[1]))))-theta28))+
(-1*((alfa2/(1+exp(-1*(delta2+gama2*t[2]))))-theta28))+
(-1*((alfa2/(1+exp(-1*(delta2+gama2*t[3]))))-theta28))+
(-1*((alfa2/(1+exp(-1*(delta2+gama2*t[4]))))-theta28))+
(-1*((alfa2/(1+exp(-1*(delta2+gama2*t[5]))))-theta28))+
(-1*((alfa2/(1+exp(-1*(delta2+gama2*t[6]))))-theta28))+
(-1*((alfa2/(1+exp(-1*(delta2+gama2*t[7]))))-theta28))+
(Yt1[i]-w8)*log((((alfa2/(1+exp(-1*(delta2+gama2*t[1]))))-theta28)))+
(Yt2[i]-w8)*log((((alfa2/(1+exp(-1*(delta2+gama2*t[2]))))-theta28)))+
(Yt3[i]-w8)*log((((alfa2/(1+exp(-1*(delta2+gama2*t[3]))))-theta28)))+
(Yt4[i]-w8)*log((((alfa2/(1+exp(-1*(delta2+gama2*t[4]))))-theta28)))+
(Yt5[i]-w8)*log((((alfa2/(1+exp(-1*(delta2+gama2*t[5]))))-theta28)))+
(Yt6[i]-w8)*log((((alfa2/(1+exp(-1*(delta2+gama2*t[6]))))-theta28)))+
(Yt7[i]-w8)*log((((alfa2/(1+exp(-1*(delta2+gama2*t[7]))))-theta28)))-
lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-
lgamma(Yt3[i]-w8+1)-lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-
lgamma(Yt6[i]-w8+1)-lgamma(Yt7[i]-w8+1))*trat2[i]+
((-1*theta38)+log(theta38**w8)+
(-1*((alfa3/(1+exp(-1*(delta3+gama3*t[1]))))-theta38))+
(-1*((alfa3/(1+exp(-1*(delta3+gama3*t[2]))))-theta38))+
(-1*((alfa3/(1+exp(-1*(delta3+gama3*t[3]))))-theta38))+
(-1*((alfa3/(1+exp(-1*(delta3+gama3*t[4]))))-theta38))+
(-1*((alfa3/(1+exp(-1*(delta3+gama3*t[5]))))-theta38))+
(-1*((alfa3/(1+exp(-1*(delta3+gama3*t[6]))))-theta38))+
(-1*((alfa3/(1+exp(-1*(delta3+gama3*t[7]))))-theta38))+
(Yt1[i]-w8)*log((((alfa3/(1+exp(-1*(delta3+gama3*t[1]))))-theta38)))+
(Yt2[i]-w8)*log((((alfa3/(1+exp(-1*(delta3+gama3*t[2]))))-theta38)))+
(Yt3[i]-w8)*log((((alfa3/(1+exp(-1*(delta3+gama3*t[3]))))-theta38)))+
(Yt4[i]-w8)*log((((alfa3/(1+exp(-1*(delta3+gama3*t[4]))))-theta38)))+
(Yt5[i]-w8)*log((((alfa3/(1+exp(-1*(delta3+gama3*t[5]))))-theta38)))+
(Yt6[i]-w8)*log((((alfa3/(1+exp(-1*(delta3+gama3*t[6]))))-theta38)))+
(Yt7[i]-w8)*log((((alfa3/(1+exp(-1*(delta3+gama3*t[7]))))-theta38)))-
lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-
lgamma(Yt3[i]-w8+1)-lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-
lgamma(Yt6[i]-w8+1)-lgamma(Yt7[i]-w8+1))*trat3[i]+
((-1*theta48)+log(theta48**w8)+
(-1*((alfa4/(1+exp(-1*(delta4+gama4*t[1]))))-theta48))+
(-1*((alfa4/(1+exp(-1*(delta4+gama4*t[2]))))-theta48))+
(-1*((alfa4/(1+exp(-1*(delta4+gama4*t[3]))))-theta48))+

```



```

(-1*((alfa4/(1+exp(-1*(delta4+gama4*t[4]))))-theta48))+
(-1*((alfa4/(1+exp(-1*(delta4+gama4*t[5]))))-theta48))+
(-1*((alfa4/(1+exp(-1*(delta4+gama4*t[6]))))-theta48))+
(-1*((alfa4/(1+exp(-1*(delta4+gama4*t[7]))))-theta48))+
(Yt1[i]-w8)*log((((alfa4/(1+exp(-1*(delta4+gama4*t[1]))))-theta48)))+
(Yt2[i]-w8)*log((((alfa4/(1+exp(-1*(delta4+gama4*t[2]))))-theta48)))+
(Yt3[i]-w8)*log((((alfa4/(1+exp(-1*(delta4+gama4*t[3]))))-theta48)))+
(Yt4[i]-w8)*log((((alfa4/(1+exp(-1*(delta4+gama4*t[4]))))-theta48)))+
(Yt5[i]-w8)*log((((alfa4/(1+exp(-1*(delta4+gama4*t[5]))))-theta48)))+
(Yt6[i]-w8)*log((((alfa4/(1+exp(-1*(delta4+gama4*t[6]))))-theta48)))+
(Yt7[i]-w8)*log((((alfa4/(1+exp(-1*(delta4+gama4*t[7]))))-theta48)))-
lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-
lgamma(Yt3[i]-w8+1)-lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-
lgamma(Yt6[i]-w8+1)-lgamma(Yt7[i]-w8+1))*trat4[i]
      p=exp(lp)
      prod=w8*p
      numer8[i]=numer8[i]+prod
      denom[i]=denom[i]+p
}
s8[i]=numer8[i]/denom[i]
s1[i]=Yt1[i]-s8[i]; s2[i]=Yt2[i]-s8[i]; s3[i]=Yt3[i]-s8[i]; s4[i]=Yt4[i]-s8[i]
s5[i]=Yt5[i]-s8[i]; s6[i]=Yt6[i]-s8[i]; s7[i]=Yt7[i]-s8[i]; lnlike[i]=log(denom[i])
}
#####Fim do Passo E#####

#####Início do Passo M#####
options(warn = 0)
out1<-optim(c(alfa1,delta1,gama1,theta18,alfa2,delta2,gama2,theta28,
             alfa3,delta3,gama3,theta38,alfa4,delta4,gama4,theta48),
            fn,sgrd,method="BFGS",hessian=TRUE)

alfa1=out1$par[1]; delta1=out1$par[2]; gama1=out1$par[3]; theta18=out1$par[4]
alfa2=out1$par[5]; delta2=out1$par[6]; gama2=out1$par[7]; theta28=out1$par[8]
alfa3=out1$par[9]; delta3=out1$par[10]; gama3=out1$par[11]; theta38=out1$par[12]
alfa4=out1$par[13]; delta4=out1$par[14]; gama4=out1$par[15]; theta48=out1$par[16]

converge=out1$convergence; names(converge)<-('converge')

m2lnL=2*out1$value; m2lnlike=-2*sum(lnlike)
difllike1=abs((m2lnL-m2lnL1)/m2lnL1); difllike2=abs((m2lnlike-m2lnL2)/m2lnL2)

names(nsimul) <- ('Simul'); names(iter)<-('iter')
printvectort1<-c(alfa1,delta1,gama1,theta18)
printvectort2<-c(alfa2,delta2,gama2,theta28)
printvectort3<-c(alfa3,delta3,gama3,theta38)

```



```

printvectort4<-c(alfa4,delta4,gama4,theta48)
printvectorvero<-c(m2lnL,difllike1,m2lnlike,difllike2)

names(printvectort1)<-c('alfa1','delta1','gama1','theta18')
names(printvectort2)<-c('alfa2','delta2','gama2','theta28')
names(printvectort3)<-c('alfa3','delta3','gama3','theta38')
names(printvectort4)<-c('alfa4','delta4','gama4','theta48')
names(printvectorvero)<-c('-2*logvero1','Rel.Dif.logvero1','-2*logvero2',
                          'Rel.Dif.logvero2')

print.default(nsimul); print.default(iter)
print.default(converge,digits=10)
print.default(printvectort1,digits=10)
print.default(printvectort2,digits=10)
print.default(printvectort3,digits=10)
print.default(printvectort4,digits=10)
print.default(printvectorvero,digits=10)
}
result[k,]=c(nsimul,iter,converge,m2lnL,difllike1,m2lnlike,difllike2,
             alfa1_ini,delta1_ini,gama1_ini,theta18_ini,
             alfa2_ini,delta2_ini,gama2_ini,theta28_ini,
             alfa3_ini,delta3_ini,gama3_ini,theta38_ini,
             alfa4_ini,delta4_ini,gama4_ini,theta48_ini)
result[k+1,]=c(nsimul,iter,converge,m2lnL,difllike1,m2lnlike,difllike2,
              alfa1,delta1,gama1,theta18,alfa2,delta2,gama2,theta28,
              alfa3,delta3,gama3,theta38,alfa4,delta4,gama4,theta48)

k=k+2
nsimul=nsimul+1
}

```

B.2 - Delineamento em blocos casualizados com efeitos fixos de bloco

```

# PARÂMETROS DA CURVA LOGÍSTICA PARA O TRATAMENTO 1
alfa1p=359.916; delta1p=-3.769; gama1p=0.022
# PARÂMETROS DA CURVA LOGÍSTICA PARA O TRATAMENTO 2
alfa2p=391.870; delta2p=-3.441; gama2p=0.021
# PARÂMETROS DA CURVA LOGÍSTICA PARA O TRATAMENTO 3
alfa3p=374.896; delta3p=-3.536; gama3p=0.022
# PARÂMETROS DA CURVA LOGÍSTICA PARA O TRATAMENTO 4
alfa4p=334.128; delta4p=-3.981; gama4p=0.026

# Cálculos dos valores sobre a curva em cada tempo para cada
# combinação de tratamento e bloco
t=c(1,46,91,136,181,226,271)

```

```

# Como função+bi >= 0, então bi>=-função
# Calculando os valores das funções para os parâmetros especificados
# Depois serão estipulados os valores iniciais para bi

fmedia = function(alfa,delta,gama,t){
  fmu=matrix(numeric(length(t)),ncol=1)
  for (i in 1:length(t)){
    fmu[i]=(alfa/(1+exp(-(delta+gama*t[i]))))
  }
  fmu
}

fmu1=fmedia(alfa1p,delta1p,gama1p,t); fmu2=fmedia(alfa2p,delta2p,gama2p,t)
fmu3=fmedia(alfa3p,delta3p,gama3p,t); fmu4=fmedia(alfa4p,delta4p,gama4p,t)

# EFEITOS DE BLOCOS
beta1p=0; beta2p=0.46; beta3p=1.85; beta4p=5.06; beta5p=2.93
betap=c(beta1p,beta2p,beta3p,beta4p,beta5p)

# CURVA MÉDIA CONSIDERANDO O EFEITO DE BLOCOS
media = function(alfa,delta,gama,t,beta){
  mu=matrix(0,nrow=5,ncol=length(t))
  for (i in 1:5){
    for (j in 1:length(t)){
      mu[i,j]=alfa/(1+exp(-(delta+gama*t[j])))+beta[i]
    }
  }
  mu
}

mu1=media(alfa1p,delta1p,gama1p,t,betap)
mu2=media(alfa2p,delta2p,gama2p,t,betap)
mu3=media(alfa3p,delta3p,gama3p,t,betap)
mu4=media(alfa4p,delta4p,gama4p,t,betap)

#####Parâmetros de Covariância
theta18p=8; theta28p=12; theta38p=10; theta48p=6

n=5
trat=c(rep(1,n),rep(2,n),rep(3,n),rep(4,n))
trat1=c(rep(1,n),rep(0,3*n))
trat2=c(rep(0,n),rep(1,n),rep(0,2*n))
trat3=c(rep(0,2*n),rep(1,n),rep(0,n))
trat4=c(rep(0,3*n),rep(1,n))
bloco=rep(1:n,4)
b2=rep(c(0,1,0,0,0),4)

```

```

b3=rep(c(0,0,1,0,0),4)
b4=rep(c(0,0,0,1,0),4)
b5=rep(c(0,0,0,0,1),4)

```

```

#### Definição da função de verossimilhança a ser maximizada #####
#### Aqui, minimiza-se -log(verossimilhança) #####
#####(-1)*log(Função de verossimilhança)#####
fn <- function(p){
lvero=(-(p[1]/(1+exp(-(p[2]+p[3]*t[1])))+p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5+
p[1]/(1+exp(-(p[2]+p[3]*t[2])))+p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5+
p[1]/(1+exp(-(p[2]+p[3]*t[3])))+p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5+
p[1]/(1+exp(-(p[2]+p[3]*t[4])))+p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5+
p[1]/(1+exp(-(p[2]+p[3]*t[5])))+p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5+
p[1]/(1+exp(-(p[2]+p[3]*t[6])))+p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5+
p[1]/(1+exp(-(p[2]+p[3]*t[7])))+p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5-
6*p[4])+log(p[4]+0.00001)*s8+
(Yt1-s8)*log(p[1]/(1+exp(-(p[2]+p[3]*t[1])))+p[17]*b2+p[18]*b3+
p[19]*b4+p[20]*b5-p[4]+0.00001)+
(Yt2-s8)*log(p[1]/(1+exp(-(p[2]+p[3]*t[2])))+p[17]*b2+p[18]*b3+
p[19]*b4+p[20]*b5-p[4]+0.00001)+
(Yt3-s8)*log(p[1]/(1+exp(-(p[2]+p[3]*t[3])))+p[17]*b2+p[18]*b3+
p[19]*b4+p[20]*b5-p[4]+0.00001)+
(Yt4-s8)*log(p[1]/(1+exp(-(p[2]+p[3]*t[4])))+p[17]*b2+p[18]*b3+
p[19]*b4+p[20]*b5-p[4]+0.00001)+
(Yt5-s8)*log(p[1]/(1+exp(-(p[2]+p[3]*t[5])))+p[17]*b2+p[18]*b3+
p[19]*b4+p[20]*b5-p[4]+0.00001)+
(Yt6-s8)*log(p[1]/(1+exp(-(p[2]+p[3]*t[6])))+p[17]*b2+p[18]*b3+
p[19]*b4+p[20]*b5-p[4]+0.00001)+
(Yt7-s8)*log(p[1]/(1+exp(-(p[2]+p[3]*t[7])))+p[17]*b2+p[18]*b3+
p[19]*b4+p[20]*b5-p[4]+0.00001)-
(lgamma(s8+1)+lgamma(Yt1-s8+1)+lgamma(Yt2-s8+1)+
lgamma(Yt3-s8+1)+lgamma(Yt4-s8+1)+lgamma(Yt5-s8+1)+
lgamma(Yt6-s8+1)+lgamma(Yt7-s8+1))*trat1+
(-(p[5]/(1+exp(-(p[6]+p[7]*t[1])))+p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5+
p[5]/(1+exp(-(p[6]+p[7]*t[2])))+p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5+
p[5]/(1+exp(-(p[6]+p[7]*t[3])))+p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5+
p[5]/(1+exp(-(p[6]+p[7]*t[4])))+p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5+
p[5]/(1+exp(-(p[6]+p[7]*t[5])))+p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5+
p[5]/(1+exp(-(p[6]+p[7]*t[6])))+p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5+
p[5]/(1+exp(-(p[6]+p[7]*t[7])))+p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5-
6*p[8])+log(p[8]+0.00001)*s8+
(Yt1-s8)*log(p[5]/(1+exp(-(p[6]+p[7]*t[1])))+p[17]*b2+p[18]*b3+
p[19]*b4+p[20]*b5-p[8]+0.00001)+
(Yt2-s8)*log(p[5]/(1+exp(-(p[6]+p[7]*t[2])))+p[17]*b2+p[18]*b3+
p[19]*b4+p[20]*b5-p[8]+0.00001)+

```

$$\begin{aligned}
& (Yt3-s8)*\log(p[5]/(1+\exp(-(p[6]+p[7]*t[3]))) + p[17]*b2+p[18]*b3+ \\
& \quad p[19]*b4+p[20]*b5-p[8]+0.00001)+ \\
& (Yt4-s8)*\log(p[5]/(1+\exp(-(p[6]+p[7]*t[4]))) + p[17]*b2+p[18]*b3+ \\
& \quad p[19]*b4+p[20]*b5-p[8]+0.00001)+ \\
& (Yt5-s8)*\log(p[5]/(1+\exp(-(p[6]+p[7]*t[5]))) + p[17]*b2+p[18]*b3+ \\
& \quad p[19]*b4+p[20]*b5-p[8]+0.00001)+ \\
& (Yt6-s8)*\log(p[5]/(1+\exp(-(p[6]+p[7]*t[6]))) + p[17]*b2+p[18]*b3+ \\
& \quad p[19]*b4+p[20]*b5-p[8]+0.00001)+ \\
& (Yt7-s8)*\log(p[5]/(1+\exp(-(p[6]+p[7]*t[7]))) + p[17]*b2+p[18]*b3+ \\
& \quad p[19]*b4+p[20]*b5-p[8]+0.00001)- \\
& (\lgamma(s8+1)+\lgamma(Yt1-s8+1)+\lgamma(Yt2-s8+1)+ \\
& \quad \lgamma(Yt3-s8+1)+\lgamma(Yt4-s8+1)+\lgamma(Yt5-s8+1)+ \\
& \quad \lgamma(Yt6-s8+1)+\lgamma(Yt7-s8+1))) *trata2+ \\
& (- (p[9]/(1+\exp(-(p[10]+p[11]*t[1]))) + p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5+ \\
& \quad p[9]/(1+\exp(-(p[10]+p[11]*t[2]))) + p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5+ \\
& \quad p[9]/(1+\exp(-(p[10]+p[11]*t[3]))) + p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5+ \\
& \quad p[9]/(1+\exp(-(p[10]+p[11]*t[4]))) + p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5+ \\
& \quad p[9]/(1+\exp(-(p[10]+p[11]*t[5]))) + p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5+ \\
& \quad p[9]/(1+\exp(-(p[10]+p[11]*t[6]))) + p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5+ \\
& \quad p[9]/(1+\exp(-(p[10]+p[11]*t[7]))) + p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5- \\
& \quad 6*p[12]) + \log(p[12]+0.00001)*s8+ \\
& (Yt1-s8)*\log(p[9]/(1+\exp(-(p[10]+p[11]*t[1]))) + p[17]*b2+p[18]*b3+ \\
& \quad p[19]*b4+p[20]*b5-p[12]+0.00001)+ \\
& (Yt2-s8)*\log(p[9]/(1+\exp(-(p[10]+p[11]*t[2]))) + p[17]*b2+p[18]*b3+ \\
& \quad p[19]*b4+p[20]*b5-p[12]+0.00001)+ \\
& (Yt3-s8)*\log(p[9]/(1+\exp(-(p[10]+p[11]*t[3]))) + p[17]*b2+p[18]*b3+ \\
& \quad p[19]*b4+p[20]*b5-p[12]+0.00001)+ \\
& (Yt4-s8)*\log(p[9]/(1+\exp(-(p[10]+p[11]*t[4]))) + p[17]*b2+p[18]*b3+ \\
& \quad p[19]*b4+p[20]*b5-p[12]+0.00001)+ \\
& (Yt5-s8)*\log(p[9]/(1+\exp(-(p[10]+p[11]*t[5]))) + p[17]*b2+p[18]*b3+ \\
& \quad p[19]*b4+p[20]*b5-p[12]+0.00001)+ \\
& (Yt6-s8)*\log(p[9]/(1+\exp(-(p[10]+p[11]*t[6]))) + p[17]*b2+p[18]*b3+ \\
& \quad p[19]*b4+p[20]*b5-p[12]+0.00001)+ \\
& (Yt7-s8)*\log(p[9]/(1+\exp(-(p[10]+p[11]*t[7]))) + p[17]*b2+p[18]*b3+ \\
& \quad p[19]*b4+p[20]*b5-p[12]+0.00001)- \\
& (\lgamma(s8+1)+\lgamma(Yt1-s8+1)+\lgamma(Yt2-s8+1)+ \\
& \quad \lgamma(Yt3-s8+1)+\lgamma(Yt4-s8+1)+\lgamma(Yt5-s8+1)+ \\
& \quad \lgamma(Yt6-s8+1)+\lgamma(Yt7-s8+1))) *trata3+ \\
& (- (p[13]/(1+\exp(-(p[14]+p[15]*t[1]))) + p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5+ \\
& \quad p[13]/(1+\exp(-(p[14]+p[15]*t[2]))) + p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5+ \\
& \quad p[13]/(1+\exp(-(p[14]+p[15]*t[3]))) + p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5+ \\
& \quad p[13]/(1+\exp(-(p[14]+p[15]*t[4]))) + p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5+ \\
& \quad p[13]/(1+\exp(-(p[14]+p[15]*t[5]))) + p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5+ \\
& \quad p[13]/(1+\exp(-(p[14]+p[15]*t[6]))) + p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5+ \\
& \quad p[13]/(1+\exp(-(p[14]+p[15]*t[7]))) + p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5-
\end{aligned}$$

```

6*p[16])+log(p[16]+0.00001)*s8+
(Yt1-s8)*log(p[13]/(1+exp(-(p[14]+p[15]*t[1])))+p[17]*b2+p[18]*b3+
p[19]*b4+p[20]*b5-p[16]+0.00001)+
(Yt2-s8)*log(p[13]/(1+exp(-(p[14]+p[15]*t[2])))+p[17]*b2+p[18]*b3+
p[19]*b4+p[20]*b5-p[16]+0.00001)+
(Yt3-s8)*log(p[13]/(1+exp(-(p[14]+p[15]*t[3])))+p[17]*b2+p[18]*b3+
p[19]*b4+p[20]*b5-p[16]+0.00001)+
(Yt4-s8)*log(p[13]/(1+exp(-(p[14]+p[15]*t[4])))+p[17]*b2+p[18]*b3+
p[19]*b4+p[20]*b5-p[16]+0.00001)+
(Yt5-s8)*log(p[13]/(1+exp(-(p[14]+p[15]*t[5])))+p[17]*b2+p[18]*b3+
p[19]*b4+p[20]*b5-p[16]+0.00001)+
(Yt6-s8)*log(p[13]/(1+exp(-(p[14]+p[15]*t[6])))+p[17]*b2+p[18]*b3+
p[19]*b4+p[20]*b5-p[16]+0.00001)+
(Yt7-s8)*log(p[13]/(1+exp(-(p[14]+p[15]*t[7])))+p[17]*b2+p[18]*b3+
p[19]*b4+p[20]*b5-p[16]+0.00001)-
(lgamma(s8+1)+lgamma(Yt1-s8+1)+lgamma(Yt2-s8+1)+
lgamma(Yt3-s8+1)+lgamma(Yt4-s8+1)+lgamma(Yt5-s8+1)+
lgamma(Yt6-s8+1)+lgamma(Yt7-s8+1))*trat4
-sum(lvero)
}

```

Definição do gradiente da função a ser minimizada

```

grd <- deriv(
  ~(-(p1/(1+exp(-(p2+p3*t1)))+p17*b2+p18*b3+p19*b4+p20*b5+
p1/(1+exp(-(p2+p3*t2)))+p17*b2+p18*b3+p19*b4+p20*b5+
p1/(1+exp(-(p2+p3*t3)))+p17*b2+p18*b3+p19*b4+p20*b5+
p1/(1+exp(-(p2+p3*t4)))+p17*b2+p18*b3+p19*b4+p20*b5+
p1/(1+exp(-(p2+p3*t5)))+p17*b2+p18*b3+p19*b4+p20*b5+
p1/(1+exp(-(p2+p3*t6)))+p17*b2+p18*b3+p19*b4+p20*b5+
p1/(1+exp(-(p2+p3*t7)))+p17*b2+p18*b3+p19*b4+p20*b5-
6*p4)+log(p4+0.00001)*s8+
(Yt1-s8)*log(p1/(1+exp(-(p2+p3*t1)))+p17*b2+p18*b3+
p19*b4+p20*b5-p4+0.00001)+
(Yt2-s8)*log(p1/(1+exp(-(p2+p3*t2)))+p17*b2+p18*b3+
p19*b4+p20*b5-p4+0.00001)+
(Yt3-s8)*log(p1/(1+exp(-(p2+p3*t3)))+p17*b2+p18*b3+
p19*b4+p20*b5-p4+0.00001)+
(Yt4-s8)*log(p1/(1+exp(-(p2+p3*t4)))+p17*b2+p18*b3+
p19*b4+p20*b5-p4+0.00001)+
(Yt5-s8)*log(p1/(1+exp(-(p2+p3*t5)))+p17*b2+p18*b3+
p19*b4+p20*b5-p4+0.00001)+
(Yt6-s8)*log(p1/(1+exp(-(p2+p3*t6)))+p17*b2+p18*b3+
p19*b4+p20*b5-p4+0.00001)+
(Yt7-s8)*log(p1/(1+exp(-(p2+p3*t7)))+p17*b2+p18*b3+
p19*b4+p20*b5-p4+0.00001)-

```

$$\begin{aligned}
& (\lgamma(s8+1)+\lgamma(Yt1-s8+1)+\lgamma(Yt2-s8+1)+ \\
& \lgamma(Yt3-s8+1)+\lgamma(Yt4-s8+1)+\lgamma(Yt5-s8+1)+ \\
& \lgamma(Yt6-s8+1)+\lgamma(Yt7-s8+1))) * \text{trat1} + \\
& (- (p5/(1+\exp(-(p6+p7*t1))))+p17*b2+p18*b3+p19*b4+p20*b5+ \\
& p5/(1+\exp(-(p6+p7*t2))))+p17*b2+p18*b3+p19*b4+p20*b5+ \\
& p5/(1+\exp(-(p6+p7*t3))))+p17*b2+p18*b3+p19*b4+p20*b5+ \\
& p5/(1+\exp(-(p6+p7*t4))))+p17*b2+p18*b3+p19*b4+p20*b5+ \\
& p5/(1+\exp(-(p6+p7*t5))))+p17*b2+p18*b3+p19*b4+p20*b5+ \\
& p5/(1+\exp(-(p6+p7*t6))))+p17*b2+p18*b3+p19*b4+p20*b5+ \\
& p5/(1+\exp(-(p6+p7*t7))))+p17*b2+p18*b3+p19*b4+p20*b5- \\
& 6*p8)+\log(p8+0.00001)*s8+ \\
& (Yt1-s8)*\log(p5/(1+\exp(-(p6+p7*t1))))+p17*b2+p18*b3+ \\
& p19*b4+p20*b5-p8+0.00001)+ \\
& (Yt2-s8)*\log(p5/(1+\exp(-(p6+p7*t2))))+p17*b2+p18*b3+ \\
& p19*b4+p20*b5-p8+0.00001)+ \\
& (Yt3-s8)*\log(p5/(1+\exp(-(p6+p7*t3))))+p17*b2+p18*b3+ \\
& p19*b4+p20*b5-p8+0.00001)+ \\
& (Yt4-s8)*\log(p5/(1+\exp(-(p6+p7*t4))))+p17*b2+p18*b3+ \\
& p19*b4+p20*b5-p8+0.00001)+ \\
& (Yt5-s8)*\log(p5/(1+\exp(-(p6+p7*t5))))+p17*b2+p18*b3+ \\
& p19*b4+p20*b5-p8+0.00001)+ \\
& (Yt6-s8)*\log(p5/(1+\exp(-(p6+p7*t6))))+p17*b2+p18*b3+ \\
& p19*b4+p20*b5-p8+0.00001)+ \\
& (Yt7-s8)*\log(p5/(1+\exp(-(p6+p7*t7))))+p17*b2+p18*b3+ \\
& p19*b4+p20*b5-p8+0.00001)- \\
& (\lgamma(s8+1)+\lgamma(Yt1-s8+1)+\lgamma(Yt2-s8+1)+ \\
& \lgamma(Yt3-s8+1)+\lgamma(Yt4-s8+1)+\lgamma(Yt5-s8+1)+ \\
& \lgamma(Yt6-s8+1)+\lgamma(Yt7-s8+1))) * \text{trat2} + \\
& (- (p9/(1+\exp(-(p10+p11*t1))))+p17*b2+p18*b3+p19*b4+p20*b5+ \\
& p9/(1+\exp(-(p10+p11*t2))))+p17*b2+p18*b3+p19*b4+p20*b5+ \\
& p9/(1+\exp(-(p10+p11*t3))))+p17*b2+p18*b3+p19*b4+p20*b5+ \\
& p9/(1+\exp(-(p10+p11*t4))))+p17*b2+p18*b3+p19*b4+p20*b5+ \\
& p9/(1+\exp(-(p10+p11*t5))))+p17*b2+p18*b3+p19*b4+p20*b5+ \\
& p9/(1+\exp(-(p10+p11*t6))))+p17*b2+p18*b3+p19*b4+p20*b5+ \\
& p9/(1+\exp(-(p10+p11*t7))))+p17*b2+p18*b3+p19*b4+p20*b5- \\
& 6*p12)+\log(p12+0.00001)*s8+ \\
& (Yt1-s8)*\log(p9/(1+\exp(-(p10+p11*t1))))+p17*b2+p18*b3+ \\
& p19*b4+p20*b5-p12+0.00001)+ \\
& (Yt2-s8)*\log(p9/(1+\exp(-(p10+p11*t2))))+p17*b2+p18*b3+ \\
& p19*b4+p20*b5-p12+0.00001)+ \\
& (Yt3-s8)*\log(p9/(1+\exp(-(p10+p11*t3))))+p17*b2+p18*b3+ \\
& p19*b4+p20*b5-p12+0.00001)+ \\
& (Yt4-s8)*\log(p9/(1+\exp(-(p10+p11*t4))))+p17*b2+p18*b3+ \\
& p19*b4+p20*b5-p12+0.00001)+ \\
& (Yt5-s8)*\log(p9/(1+\exp(-(p10+p11*t5))))+p17*b2+p18*b3+
\end{aligned}$$

```

p19*b4+p20*b5-p12+0.00001)+
(Yt6-s8)*log(p9/(1+exp(-(p10+p11*t6))))+p17*b2+p18*b3+
p19*b4+p20*b5-p12+0.00001)+
(Yt7-s8)*log(p9/(1+exp(-(p10+p11*t7))))+p17*b2+p18*b3+
p19*b4+p20*b5-p12+0.00001)-
(lgamma(s8+1)+lgamma(Yt1-s8+1)+lgamma(Yt2-s8+1)+
lgamma(Yt3-s8+1)+lgamma(Yt4-s8+1)+lgamma(Yt5-s8+1)+
lgamma(Yt6-s8+1)+lgamma(Yt7-s8+1)))*trat3+
(-(p13/(1+exp(-(p14+p15*t1))))+p17*b2+p18*b3+p19*b4+p20*b5+
p13/(1+exp(-(p14+p15*t2))))+p17*b2+p18*b3+p19*b4+p20*b5+
p13/(1+exp(-(p14+p15*t3))))+p17*b2+p18*b3+p19*b4+p20*b5+
p13/(1+exp(-(p14+p15*t4))))+p17*b2+p18*b3+p19*b4+p20*b5+
p13/(1+exp(-(p14+p15*t5))))+p17*b2+p18*b3+p19*b4+p20*b5+
p13/(1+exp(-(p14+p15*t6))))+p17*b2+p18*b3+p19*b4+p20*b5+
p13/(1+exp(-(p14+p15*t7))))+p17*b2+p18*b3+p19*b4+p20*b5-
6*p16)+log(p16+0.00001)*s8+
(Yt1-s8)*log(p13/(1+exp(-(p14+p15*t1))))+p17*b2+p18*b3+
p19*b4+p20*b5-p16+0.00001)+
(Yt2-s8)*log(p13/(1+exp(-(p14+p15*t2))))+p17*b2+p18*b3+
p19*b4+p20*b5-p16+0.00001)+
(Yt3-s8)*log(p13/(1+exp(-(p14+p15*t3))))+p17*b2+p18*b3+
p19*b4+p20*b5-p16+0.00001)+
(Yt4-s8)*log(p13/(1+exp(-(p14+p15*t4))))+p17*b2+p18*b3+
p19*b4+p20*b5-p16+0.00001)+
(Yt5-s8)*log(p13/(1+exp(-(p14+p15*t5))))+p17*b2+p18*b3+
p19*b4+p20*b5-p16+0.00001)+
(Yt6-s8)*log(p13/(1+exp(-(p14+p15*t6))))+p17*b2+p18*b3+
p19*b4+p20*b5-p16+0.00001)+
(Yt7-s8)*log(p13/(1+exp(-(p14+p15*t7))))+p17*b2+p18*b3+
p19*b4+p20*b5-p16+0.00001)-
(lgamma(s8+1)+lgamma(Yt1-s8+1)+lgamma(Yt2-s8+1)+
lgamma(Yt3-s8+1)+lgamma(Yt4-s8+1)+lgamma(Yt5-s8+1)+
lgamma(Yt6-s8+1)+lgamma(Yt7-s8+1)))*trat4,
c("p1", "p2", "p3", "p4", "p5", "p6", "p7", "p8", "p9", "p10", "p11", "p12",
  "p13", "p14", "p15", "p16", "p17", "p18", "p19", "p20"),
function(p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11,p12,p13,p14,p15,
  p16,p17,p18,p19,p20,t1,t2,t3,t4,t5,t6,t7) NULL)

```

```

sgrd <- function(p,Yt1,Yt2,Yt3,Yt4,Yt5,Yt6,Yt7,s8) {
  p1=p[1]; p2=p[2]; p3=p[3]; p4=p[4]; p5=p[5]; p6=p[6]; p7=p[7]
  p8=p[8]; p9=p[9]; p10=p[10]; p11=p[11]; p12=p[12]; p13=p[13]
  p14=p[14]; p15=p[15]; p16=p[16]; p17=p[17]; p18=p[18]; p19=p[19]
  p20=p[20]
  t1=t[1]; t2=t[2]; t3=t[3]; t4=t[4]; t5=t[5]; t6=t[6]; t7=t[7]
  colSums(attr(grd(p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11,p12,p13,p14,p15,

```



```

        p16,p17,p18,p19,p20,t1,t2,t3,t4,t5,t6,t7),"gradient"))
    }

##### Início do Processo de Simulação #####

library(nlreg)

W11=matrix(numeric(length(betap)),1); W12=matrix(numeric(length(betap)),1)
W13=matrix(numeric(length(betap)),1); W14=matrix(numeric(length(betap)),1)
W15=matrix(numeric(length(betap)),1); W16=matrix(numeric(length(betap)),1)
W17=matrix(numeric(length(betap)),1)

W21=matrix(numeric(length(betap)),1); W22=matrix(numeric(length(betap)),1)
W23=matrix(numeric(length(betap)),1); W24=matrix(numeric(length(betap)),1)
W25=matrix(numeric(length(betap)),1); W26=matrix(numeric(length(betap)),1)
W27=matrix(numeric(length(betap)),1)

W31=matrix(numeric(length(betap)),1); W32=matrix(numeric(length(betap)),1)
W33=matrix(numeric(length(betap)),1); W34=matrix(numeric(length(betap)),1)
W35=matrix(numeric(length(betap)),1); W36=matrix(numeric(length(betap)),1)
W37=matrix(numeric(length(betap)),1)

W41=matrix(numeric(length(betap)),1); W42=matrix(numeric(length(betap)),1)
W43=matrix(numeric(length(betap)),1); W44=matrix(numeric(length(betap)),1)
W45=matrix(numeric(length(betap)),1); W46=matrix(numeric(length(betap)),1)
W47=matrix(numeric(length(betap)),1)

set.seed(33); smt=runif(1000)
smt=round(smt,digits=5); smt=smt*100000
result=matrix(0,nrow=2000,ncol=28)
k=1; nsimul=1
while (nsimul <= 1000){
  set.seed(smt[nsimul])

  for (i in 1:length(betap)){
    #Variáveis latentes para o tratamento 1
    W11[i]=rpois(1,fmu1[1]+betap[i]-theta18p)
    W12[i]=rpois(1,fmu1[2]+betap[i]-theta18p)
    W13[i]=rpois(1,fmu1[3]+betap[i]-theta18p)
    W14[i]=rpois(1,fmu1[4]+betap[i]-theta18p)
    W15[i]=rpois(1,fmu1[5]+betap[i]-theta18p)
    W16[i]=rpois(1,fmu1[6]+betap[i]-theta18p)
    W17[i]=rpois(1,fmu1[7]+betap[i]-theta18p)

    #Variáveis latentes para o tratamento 2

```



```

W21[i]=rpois(1,fmu2[1]+betap[i]-theta28p)
W22[i]=rpois(1,fmu2[2]+betap[i]-theta28p)
W23[i]=rpois(1,fmu2[3]+betap[i]-theta28p)
W24[i]=rpois(1,fmu2[4]+betap[i]-theta28p)
W25[i]=rpois(1,fmu2[5]+betap[i]-theta28p)
W26[i]=rpois(1,fmu2[6]+betap[i]-theta28p)
W27[i]=rpois(1,fmu2[7]+betap[i]-theta28p)

#Variáveis latentes para o tratamento 3
W31[i]=rpois(1,fmu3[1]+betap[i]-theta38p)
W32[i]=rpois(1,fmu3[2]+betap[i]-theta38p)
W33[i]=rpois(1,fmu3[3]+betap[i]-theta38p)
W34[i]=rpois(1,fmu3[4]+betap[i]-theta38p)
W35[i]=rpois(1,fmu3[5]+betap[i]-theta38p)
W36[i]=rpois(1,fmu3[6]+betap[i]-theta38p)
W37[i]=rpois(1,fmu3[7]+betap[i]-theta38p)

#Variáveis latentes para o tratamento 4
W41[i]=rpois(1,fmu4[1]+betap[i]-theta48p)
W42[i]=rpois(1,fmu4[2]+betap[i]-theta48p)
W43[i]=rpois(1,fmu4[3]+betap[i]-theta48p)
W44[i]=rpois(1,fmu4[4]+betap[i]-theta48p)
W45[i]=rpois(1,fmu4[5]+betap[i]-theta48p)
W46[i]=rpois(1,fmu4[6]+betap[i]-theta48p)
W47[i]=rpois(1,fmu4[7]+betap[i]-theta48p)
}

W18=rpois(n,theta18p); W28=rpois(n,theta28p)
W38=rpois(n,theta38p); W48=rpois(n,theta48p)

#Variáveis observadas para o tratamento 1
Y11=W11+W18; Y12=W12+W18; Y13=W13+W18; Y14=W14+W18
Y15=W15+W18; Y16=W16+W18; Y17=W17+W18;

Y1=cbind(t(Y11),t(Y12),t(Y13),t(Y14),t(Y15),t(Y16),t(Y17))
Ytrat1=rbind(t(Y11),t(Y12),t(Y13),t(Y14),t(Y15),t(Y16),t(Y17))
Ytrat1.reg=rbind(t(Y11)[1],t(Y12)[1],t(Y13)[1],t(Y14)[1],
                 t(Y15)[1],t(Y16)[1],t(Y17)[1])

#Variáveis observadas para o tratamento 2
Y21=W21+W28; Y22=W22+W28; Y23=W23+W28; Y24=W24+W28
Y25=W25+W28; Y26=W26+W28; Y27=W27+W28;

Y2=cbind(t(Y21),t(Y22),t(Y23),t(Y24),t(Y25),t(Y26),t(Y27))
Ytrat2=rbind(t(Y21),t(Y22),t(Y23),t(Y24),t(Y25),t(Y26),t(Y27))

```

```

Ytrat2.reg=rbind(t(Y21)[1],t(Y22)[1],t(Y23)[1],t(Y24)[1],
                t(Y25)[1],t(Y26)[1],t(Y27)[1])

#Variáveis observadas para o tratamento 3
Y31=W31+W38; Y32=W32+W38; Y33=W33+W38; Y34=W34+W38
Y35=W35+W38; Y36=W36+W38; Y37=W37+W38

Y3=cbind(t(Y31),t(Y32),t(Y33),t(Y34),t(Y35),t(Y36),t(Y37))
Ytrat3=rbind(t(Y31),t(Y32),t(Y33),t(Y34),t(Y35),t(Y36),t(Y37))
Ytrat3.reg=rbind(t(Y31)[1],t(Y32)[1],t(Y33)[1],t(Y34)[1],
                t(Y35)[1],t(Y36)[1],t(Y37)[1])

#Variáveis observadas para o tratamento 4
Y41=W41+W48; Y42=W42+W48; Y43=W43+W48; Y44=W44+W48
Y45=W45+W48; Y46=W46+W48; Y47=W47+W48

Y4=cbind(t(Y41),t(Y42),t(Y43),t(Y44),t(Y45),t(Y46),t(Y47))
Ytrat4=rbind(t(Y41),t(Y42),t(Y43),t(Y44),t(Y45),t(Y46),t(Y47))
Ytrat4.reg=rbind(t(Y41)[1],t(Y42)[1],t(Y43)[1],t(Y44)[1],
                t(Y45)[1],t(Y46)[1],t(Y47)[1])

Y=rbind(Y1,Y2,Y3,Y4)
Yt1=Y[,1]; Yt2=Y[,2]; Yt3=Y[,3]; Yt4=Y[,4]
Yt5=Y[,5]; Yt6=Y[,6]; Yt7=Y[,7]

tempo=c(rep(t[1],n),rep(t[2],n),rep(t[3],n),rep(t[4],n),rep(t[5],n),
        rep(t[6],n),rep(t[7],n))
Ytrat=data.frame(tempo,Ytrat1,Ytrat2,Ytrat3,Ytrat4)
Ytrat.reg=data.frame(t,Ytrat1.reg,Ytrat2.reg,Ytrat3.reg,Ytrat4.reg)

#Encontrando valores iniciais para os parâmetros da regressão
init1 <- nlreg( Ytrat1.reg ~ (alfa1/(1+exp(-1*(delta1+gama1*t)))),
               start = c(alfa1=100, delta1=-1, gama1=0.05), data = Ytrat.reg)
init2 <- nlreg( Ytrat2.reg ~ (alfa2/(1+exp(-1*(delta2+gama2*t)))),
               start = c(alfa2=100, delta2=-1, gama2=0.05), data = Ytrat.reg)
init3 <- nlreg( Ytrat3.reg ~ (alfa3/(1+exp(-1*(delta3+gama3*t)))),
               start = c(alfa3=100, delta3=-1, gama3=0.05), data = Ytrat.reg)
init4 <- nlreg( Ytrat4.reg ~ (alfa4/(1+exp(-1*(delta4+gama4*t)))),
               start = c(alfa4=100, delta4=-1, gama4=0.05), data = Ytrat.reg)

alfa1=init1$coef[1]; delta1=init1$coef[2]; gama1=init1$coef[3]
alfa1_ini=alfa1; delta1_ini=delta1; gama1_ini=gama1
alfa2=init2$coef[1]; delta2=init2$coef[2]; gama2=init2$coef[3]
alfa2_ini=alfa2; delta2_ini=delta2; gama2_ini=gama2
alfa3=init3$coef[1]; delta3=init3$coef[2]; gama3=init3$coef[3]

```

```

alfa3_ini=alfa3; delta3_ini=delta3; gama3_ini=gama3
alfa4=init4$coef[1]; delta4=init4$coef[2]; gama4=init4$coef[3]
alfa4_ini=alfa4; delta4_ini=delta4; gama4_ini=gama4

# Procedimento para estimar o efeito de bloco para usar como chute inicial
Yt1.bloco=cbind(trat,bloco,Yt1); Yt2.bloco=cbind(trat,bloco,Yt2)
Yt3.bloco=cbind(trat,bloco,Yt3); Yt4.bloco=cbind(trat,bloco,Yt4)
Yt5.bloco=cbind(trat,bloco,Yt5); Yt6.bloco=cbind(trat,bloco,Yt6)
Yt7.bloco=cbind(trat,bloco,Yt7)

dados.bloco=data.frame(rbind(Yt1.bloco,Yt2.bloco,Yt3.bloco,
                             Yt4.bloco,Yt5.bloco,Yt6.bloco,Yt7.bloco))

tp=c(rep(1,20),rep(2,20),rep(3,20),rep(4,20),rep(5,20),rep(6,20),rep(7,20))

dados.bloco$tp=tp
init.bloco = glm(Yt1 ~ factor(trat)+factor(bloco),data=dados.bloco)
b=c(0,init.bloco$coef[5:8])

beta1=b[1]; beta1_ini=beta1

beta2=max(b[2],-min(alfa1/(1+exp(-1*(delta1+gama1*t[1]))),
                    alfa1/(1+exp(-1*(delta1+gama1*t[2]))),
                    alfa1/(1+exp(-1*(delta1+gama1*t[3]))),
                    alfa1/(1+exp(-1*(delta1+gama1*t[4]))),
                    alfa1/(1+exp(-1*(delta1+gama1*t[5]))),
                    alfa1/(1+exp(-1*(delta1+gama1*t[6]))),
                    alfa1/(1+exp(-1*(delta1+gama1*t[7]))),
                    alfa2/(1+exp(-1*(delta2+gama2*t[1]))),
                    alfa2/(1+exp(-1*(delta2+gama2*t[2]))),
                    alfa2/(1+exp(-1*(delta2+gama2*t[3]))),
                    alfa2/(1+exp(-1*(delta2+gama2*t[4]))),
                    alfa2/(1+exp(-1*(delta2+gama2*t[5]))),
                    alfa2/(1+exp(-1*(delta2+gama2*t[6]))),
                    alfa2/(1+exp(-1*(delta2+gama2*t[7]))),
                    alfa3/(1+exp(-1*(delta3+gama3*t[1]))),
                    alfa3/(1+exp(-1*(delta3+gama3*t[2]))),
                    alfa3/(1+exp(-1*(delta3+gama3*t[3]))),
                    alfa3/(1+exp(-1*(delta3+gama3*t[4]))),
                    alfa3/(1+exp(-1*(delta3+gama3*t[5]))),
                    alfa3/(1+exp(-1*(delta3+gama3*t[6]))),
                    alfa3/(1+exp(-1*(delta3+gama3*t[7]))),
                    alfa4/(1+exp(-1*(delta4+gama4*t[1]))),
                    alfa4/(1+exp(-1*(delta4+gama4*t[2]))),
                    alfa4/(1+exp(-1*(delta4+gama4*t[3]))),

```

```

        alfa4/(1+exp(-1*(delta4+gama4*t[4]))),
        alfa4/(1+exp(-1*(delta4+gama4*t[5]))),
        alfa4/(1+exp(-1*(delta4+gama4*t[6]))),
        alfa4/(1+exp(-1*(delta4+gama4*t[7]))))
beta2_ini=beta2

beta3=max(b[3],-min(alfa1/(1+exp(-1*(delta1+gama1*t[1]))),
        alfa1/(1+exp(-1*(delta1+gama1*t[2]))),
        alfa1/(1+exp(-1*(delta1+gama1*t[3]))),
        alfa1/(1+exp(-1*(delta1+gama1*t[4]))),
        alfa1/(1+exp(-1*(delta1+gama1*t[5]))),
        alfa1/(1+exp(-1*(delta1+gama1*t[6]))),
        alfa1/(1+exp(-1*(delta1+gama1*t[7]))),
        alfa2/(1+exp(-1*(delta2+gama2*t[1]))),
        alfa2/(1+exp(-1*(delta2+gama2*t[2]))),
        alfa2/(1+exp(-1*(delta2+gama2*t[3]))),
        alfa2/(1+exp(-1*(delta2+gama2*t[4]))),
        alfa2/(1+exp(-1*(delta2+gama2*t[5]))),
        alfa2/(1+exp(-1*(delta2+gama2*t[6]))),
        alfa2/(1+exp(-1*(delta2+gama2*t[7]))),
        alfa3/(1+exp(-1*(delta3+gama3*t[1]))),
        alfa3/(1+exp(-1*(delta3+gama3*t[2]))),
        alfa3/(1+exp(-1*(delta3+gama3*t[3]))),
        alfa3/(1+exp(-1*(delta3+gama3*t[4]))),
        alfa3/(1+exp(-1*(delta3+gama3*t[5]))),
        alfa3/(1+exp(-1*(delta3+gama3*t[6]))),
        alfa3/(1+exp(-1*(delta3+gama3*t[7]))),
        alfa4/(1+exp(-1*(delta4+gama4*t[1]))),
        alfa4/(1+exp(-1*(delta4+gama4*t[2]))),
        alfa4/(1+exp(-1*(delta4+gama4*t[3]))),
        alfa4/(1+exp(-1*(delta4+gama4*t[4]))),
        alfa4/(1+exp(-1*(delta4+gama4*t[5]))),
        alfa4/(1+exp(-1*(delta4+gama4*t[6]))),
        alfa4/(1+exp(-1*(delta4+gama4*t[7]))))
beta3_ini=beta3

beta4=max(b[4],-min(alfa1/(1+exp(-1*(delta1+gama1*t[1]))),
        alfa1/(1+exp(-1*(delta1+gama1*t[2]))),
        alfa1/(1+exp(-1*(delta1+gama1*t[3]))),
        alfa1/(1+exp(-1*(delta1+gama1*t[4]))),
        alfa1/(1+exp(-1*(delta1+gama1*t[5]))),
        alfa1/(1+exp(-1*(delta1+gama1*t[6]))),
        alfa1/(1+exp(-1*(delta1+gama1*t[7]))),
        alfa2/(1+exp(-1*(delta2+gama2*t[1]))),
        alfa2/(1+exp(-1*(delta2+gama2*t[2]))),

```

```

    alfa2/(1+exp(-1*(delta2+gama2*t[3]))),
    alfa2/(1+exp(-1*(delta2+gama2*t[4]))),
    alfa2/(1+exp(-1*(delta2+gama2*t[5]))),
    alfa2/(1+exp(-1*(delta2+gama2*t[6]))),
    alfa2/(1+exp(-1*(delta2+gama2*t[7]))),
    alfa3/(1+exp(-1*(delta3+gama3*t[1]))),
    alfa3/(1+exp(-1*(delta3+gama3*t[2]))),
    alfa3/(1+exp(-1*(delta3+gama3*t[3]))),
    alfa3/(1+exp(-1*(delta3+gama3*t[4]))),
    alfa3/(1+exp(-1*(delta3+gama3*t[5]))),
    alfa3/(1+exp(-1*(delta3+gama3*t[6]))),
    alfa3/(1+exp(-1*(delta3+gama3*t[7]))),
    alfa4/(1+exp(-1*(delta4+gama4*t[1]))),
    alfa4/(1+exp(-1*(delta4+gama4*t[2]))),
    alfa4/(1+exp(-1*(delta4+gama4*t[3]))),
    alfa4/(1+exp(-1*(delta4+gama4*t[4]))),
    alfa4/(1+exp(-1*(delta4+gama4*t[5]))),
    alfa4/(1+exp(-1*(delta4+gama4*t[6]))),
    alfa4/(1+exp(-1*(delta4+gama4*t[7]))))
beta4_ini=beta4

beta5=max(b[5],-min(alfa1/(1+exp(-1*(delta1+gama1*t[1]))),
    alfa1/(1+exp(-1*(delta1+gama1*t[2]))),
    alfa1/(1+exp(-1*(delta1+gama1*t[3]))),
    alfa1/(1+exp(-1*(delta1+gama1*t[4]))),
    alfa1/(1+exp(-1*(delta1+gama1*t[5]))),
    alfa1/(1+exp(-1*(delta1+gama1*t[6]))),
    alfa1/(1+exp(-1*(delta1+gama1*t[7]))),
    alfa2/(1+exp(-1*(delta2+gama2*t[1]))),
    alfa2/(1+exp(-1*(delta2+gama2*t[2]))),
    alfa2/(1+exp(-1*(delta2+gama2*t[3]))),
    alfa2/(1+exp(-1*(delta2+gama2*t[4]))),
    alfa2/(1+exp(-1*(delta2+gama2*t[5]))),
    alfa2/(1+exp(-1*(delta2+gama2*t[6]))),
    alfa2/(1+exp(-1*(delta2+gama2*t[7]))),
    alfa3/(1+exp(-1*(delta3+gama3*t[1]))),
    alfa3/(1+exp(-1*(delta3+gama3*t[2]))),
    alfa3/(1+exp(-1*(delta3+gama3*t[3]))),
    alfa3/(1+exp(-1*(delta3+gama3*t[4]))),
    alfa3/(1+exp(-1*(delta3+gama3*t[5]))),
    alfa3/(1+exp(-1*(delta3+gama3*t[6]))),
    alfa3/(1+exp(-1*(delta3+gama3*t[7]))),
    alfa4/(1+exp(-1*(delta4+gama4*t[1]))),
    alfa4/(1+exp(-1*(delta4+gama4*t[2]))),
    alfa4/(1+exp(-1*(delta4+gama4*t[3]))),

```

```

    alfa4/(1+exp(-1*(delta4+gama4*t[4]))),
    alfa4/(1+exp(-1*(delta4+gama4*t[5]))),
    alfa4/(1+exp(-1*(delta4+gama4*t[6]))),
    alfa4/(1+exp(-1*(delta4+gama4*t[7]))))
beta5_ini=beta5

#Encontrando os valores iniciais para os parâmetros de covariância

if (cov(t(Y11),t(Y12))>0) cov12_1=cov(t(Y11),t(Y12)) else cov12_1=0
if (cov(t(Y11),t(Y13))>0) cov13_1=cov(t(Y11),t(Y13)) else cov13_1=0
if (cov(t(Y11),t(Y14))>0) cov14_1=cov(t(Y11),t(Y14)) else cov14_1=0
if (cov(t(Y11),t(Y15))>0) cov15_1=cov(t(Y11),t(Y15)) else cov15_1=0
if (cov(t(Y11),t(Y16))>0) cov16_1=cov(t(Y11),t(Y16)) else cov16_1=0
if (cov(t(Y11),t(Y17))>0) cov17_1=cov(t(Y11),t(Y17)) else cov17_1=0
if (cov(t(Y12),t(Y13))>0) cov23_1=cov(t(Y12),t(Y13)) else cov23_1=0
if (cov(t(Y12),t(Y14))>0) cov24_1=cov(t(Y12),t(Y14)) else cov24_1=0
if (cov(t(Y12),t(Y15))>0) cov25_1=cov(t(Y12),t(Y15)) else cov25_1=0
if (cov(t(Y12),t(Y16))>0) cov26_1=cov(t(Y12),t(Y16)) else cov26_1=0
if (cov(t(Y12),t(Y17))>0) cov27_1=cov(t(Y12),t(Y17)) else cov27_1=0
if (cov(t(Y13),t(Y14))>0) cov34_1=cov(t(Y13),t(Y14)) else cov34_1=0
if (cov(t(Y13),t(Y15))>0) cov35_1=cov(t(Y13),t(Y15)) else cov35_1=0
if (cov(t(Y13),t(Y16))>0) cov36_1=cov(t(Y13),t(Y16)) else cov36_1=0
if (cov(t(Y13),t(Y17))>0) cov37_1=cov(t(Y13),t(Y17)) else cov37_1=0
if (cov(t(Y14),t(Y15))>0) cov45_1=cov(t(Y14),t(Y15)) else cov45_1=0
if (cov(t(Y14),t(Y16))>0) cov46_1=cov(t(Y14),t(Y16)) else cov46_1=0
if (cov(t(Y14),t(Y17))>0) cov47_1=cov(t(Y14),t(Y17)) else cov47_1=0
if (cov(t(Y15),t(Y16))>0) cov56_1=cov(t(Y15),t(Y16)) else cov56_1=0
if (cov(t(Y15),t(Y17))>0) cov57_1=cov(t(Y15),t(Y17)) else cov57_1=0
if (cov(t(Y16),t(Y17))>0) cov67_1=cov(t(Y16),t(Y17)) else cov67_1=0

cov1=c(cov12_1,cov13_1,cov14_1,cov15_1,cov16_1,cov17_1,cov23_1,cov24_1,
      cov25_1,cov26_1,cov27_1,cov34_1,cov35_1,cov36_1,cov37_1,cov45_1,
      cov46_1,cov47_1,cov56_1,cov57_1,cov67_1)

theta18=min(mean(cov1),(alfa1/(1+exp(-1*(delta1+gama1*t[1]))))+beta1,
            (alfa1/(1+exp(-1*(delta1+gama1*t[1]))))+beta2,
            (alfa1/(1+exp(-1*(delta1+gama1*t[1]))))+beta3,
            (alfa1/(1+exp(-1*(delta1+gama1*t[1]))))+beta4,
            (alfa1/(1+exp(-1*(delta1+gama1*t[1]))))+beta5,
            (alfa1/(1+exp(-1*(delta1+gama1*t[2]))))+beta1,
            (alfa1/(1+exp(-1*(delta1+gama1*t[2]))))+beta2,
            (alfa1/(1+exp(-1*(delta1+gama1*t[2]))))+beta3,
            (alfa1/(1+exp(-1*(delta1+gama1*t[2]))))+beta4,
            (alfa1/(1+exp(-1*(delta1+gama1*t[2]))))+beta5,
            (alfa1/(1+exp(-1*(delta1+gama1*t[3]))))+beta1,

```

```

(alfa1/(1+exp(-1*(delta1+gama1*t[3]))))+beta2,
(alfa1/(1+exp(-1*(delta1+gama1*t[3]))))+beta3,
(alfa1/(1+exp(-1*(delta1+gama1*t[3]))))+beta4,
(alfa1/(1+exp(-1*(delta1+gama1*t[3]))))+beta5,
(alfa1/(1+exp(-1*(delta1+gama1*t[4]))))+beta1,
(alfa1/(1+exp(-1*(delta1+gama1*t[4]))))+beta2,
(alfa1/(1+exp(-1*(delta1+gama1*t[4]))))+beta3,
(alfa1/(1+exp(-1*(delta1+gama1*t[4]))))+beta4,
(alfa1/(1+exp(-1*(delta1+gama1*t[4]))))+beta5,
(alfa1/(1+exp(-1*(delta1+gama1*t[5]))))+beta1,
(alfa1/(1+exp(-1*(delta1+gama1*t[5]))))+beta2,
(alfa1/(1+exp(-1*(delta1+gama1*t[5]))))+beta3,
(alfa1/(1+exp(-1*(delta1+gama1*t[5]))))+beta4,
(alfa1/(1+exp(-1*(delta1+gama1*t[5]))))+beta5,
(alfa1/(1+exp(-1*(delta1+gama1*t[6]))))+beta1,
(alfa1/(1+exp(-1*(delta1+gama1*t[6]))))+beta2,
(alfa1/(1+exp(-1*(delta1+gama1*t[6]))))+beta3,
(alfa1/(1+exp(-1*(delta1+gama1*t[6]))))+beta4,
(alfa1/(1+exp(-1*(delta1+gama1*t[6]))))+beta5,
(alfa1/(1+exp(-1*(delta1+gama1*t[7]))))+beta1,
(alfa1/(1+exp(-1*(delta1+gama1*t[7]))))+beta2,
(alfa1/(1+exp(-1*(delta1+gama1*t[7]))))+beta3,
(alfa1/(1+exp(-1*(delta1+gama1*t[7]))))+beta4,
(alfa1/(1+exp(-1*(delta1+gama1*t[7]))))+beta5)
theta18_ini=theta18

if (cov(t(Y21),t(Y22))>0) cov12_2=cov(t(Y21),t(Y22)) else cov12_2=0
if (cov(t(Y21),t(Y23))>0) cov13_2=cov(t(Y21),t(Y23)) else cov13_2=0
if (cov(t(Y21),t(Y24))>0) cov14_2=cov(t(Y21),t(Y24)) else cov14_2=0
if (cov(t(Y21),t(Y25))>0) cov15_2=cov(t(Y21),t(Y25)) else cov15_2=0
if (cov(t(Y21),t(Y26))>0) cov16_2=cov(t(Y21),t(Y26)) else cov16_2=0
if (cov(t(Y21),t(Y27))>0) cov17_2=cov(t(Y21),t(Y27)) else cov17_2=0
if (cov(t(Y22),t(Y23))>0) cov23_2=cov(t(Y22),t(Y23)) else cov23_2=0
if (cov(t(Y22),t(Y24))>0) cov24_2=cov(t(Y22),t(Y24)) else cov24_2=0
if (cov(t(Y22),t(Y25))>0) cov25_2=cov(t(Y22),t(Y25)) else cov25_2=0
if (cov(t(Y22),t(Y26))>0) cov26_2=cov(t(Y22),t(Y26)) else cov26_2=0
if (cov(t(Y22),t(Y27))>0) cov27_2=cov(t(Y22),t(Y27)) else cov27_2=0
if (cov(t(Y23),t(Y24))>0) cov34_2=cov(t(Y23),t(Y24)) else cov34_2=0
if (cov(t(Y23),t(Y25))>0) cov35_2=cov(t(Y23),t(Y25)) else cov35_2=0
if (cov(t(Y23),t(Y26))>0) cov36_2=cov(t(Y23),t(Y26)) else cov36_2=0
if (cov(t(Y23),t(Y27))>0) cov37_2=cov(t(Y23),t(Y27)) else cov37_2=0
if (cov(t(Y24),t(Y25))>0) cov45_2=cov(t(Y24),t(Y25)) else cov45_2=0
if (cov(t(Y24),t(Y26))>0) cov46_2=cov(t(Y24),t(Y26)) else cov46_2=0
if (cov(t(Y24),t(Y27))>0) cov47_2=cov(t(Y24),t(Y27)) else cov47_2=0
if (cov(t(Y25),t(Y26))>0) cov56_2=cov(t(Y25),t(Y26)) else cov56_2=0

```



```

if (cov(t(Y25),t(Y27))>0) cov57_2=cov(t(Y25),t(Y27)) else cov57_2=0
if (cov(t(Y26),t(Y27))>0) cov67_2=cov(t(Y26),t(Y27)) else cov67_2=0

cov2=c(cov12_2,cov13_2,cov14_2,cov15_2,cov16_2,cov17_2,cov23_2,cov24_2,
      cov25_2,cov26_2,cov27_2,cov34_2,cov35_2,cov36_2,cov37_2,cov45_2,
      cov46_2,cov47_2,cov56_2,cov57_2,cov67_2)

theta28=min(mean(cov2),(alfa2/(1+exp(-1*(delta2+gama2*t[1]))))+beta1,
            (alfa2/(1+exp(-1*(delta2+gama2*t[1]))))+beta2,
            (alfa2/(1+exp(-1*(delta2+gama2*t[1]))))+beta3,
            (alfa2/(1+exp(-1*(delta2+gama2*t[1]))))+beta4,
            (alfa2/(1+exp(-1*(delta2+gama2*t[1]))))+beta5,
            (alfa2/(1+exp(-1*(delta2+gama2*t[2]))))+beta1,
            (alfa2/(1+exp(-1*(delta2+gama2*t[2]))))+beta2,
            (alfa2/(1+exp(-1*(delta2+gama2*t[2]))))+beta3,
            (alfa2/(1+exp(-1*(delta2+gama2*t[2]))))+beta4,
            (alfa2/(1+exp(-1*(delta2+gama2*t[2]))))+beta5,
            (alfa2/(1+exp(-1*(delta2+gama2*t[3]))))+beta1,
            (alfa2/(1+exp(-1*(delta2+gama2*t[3]))))+beta2,
            (alfa2/(1+exp(-1*(delta2+gama2*t[3]))))+beta3,
            (alfa2/(1+exp(-1*(delta2+gama2*t[3]))))+beta4,
            (alfa2/(1+exp(-1*(delta2+gama2*t[3]))))+beta5,
            (alfa2/(1+exp(-1*(delta2+gama2*t[4]))))+beta1,
            (alfa2/(1+exp(-1*(delta2+gama2*t[4]))))+beta2,
            (alfa2/(1+exp(-1*(delta2+gama2*t[4]))))+beta3,
            (alfa2/(1+exp(-1*(delta2+gama2*t[4]))))+beta4,
            (alfa2/(1+exp(-1*(delta2+gama2*t[4]))))+beta5,
            (alfa2/(1+exp(-1*(delta2+gama2*t[5]))))+beta1,
            (alfa2/(1+exp(-1*(delta2+gama2*t[5]))))+beta2,
            (alfa2/(1+exp(-1*(delta2+gama2*t[5]))))+beta3,
            (alfa2/(1+exp(-1*(delta2+gama2*t[5]))))+beta4,
            (alfa2/(1+exp(-1*(delta2+gama2*t[5]))))+beta5,
            (alfa2/(1+exp(-1*(delta2+gama2*t[6]))))+beta1,
            (alfa2/(1+exp(-1*(delta2+gama2*t[6]))))+beta2,
            (alfa2/(1+exp(-1*(delta2+gama2*t[6]))))+beta3,
            (alfa2/(1+exp(-1*(delta2+gama2*t[6]))))+beta4,
            (alfa2/(1+exp(-1*(delta2+gama2*t[6]))))+beta5,
            (alfa2/(1+exp(-1*(delta2+gama2*t[7]))))+beta1,
            (alfa2/(1+exp(-1*(delta2+gama2*t[7]))))+beta2,
            (alfa2/(1+exp(-1*(delta2+gama2*t[7]))))+beta3,
            (alfa2/(1+exp(-1*(delta2+gama2*t[7]))))+beta4,
            (alfa2/(1+exp(-1*(delta2+gama2*t[7]))))+beta5)
theta28_ini=theta28

if (cov(t(Y31),t(Y32))>0) cov12_3=cov(t(Y31),t(Y32)) else cov12_3=0

```



```

if (cov(t(Y31),t(Y33))>0) cov13_3=cov(t(Y31),t(Y33)) else cov13_3=0
if (cov(t(Y31),t(Y34))>0) cov14_3=cov(t(Y31),t(Y34)) else cov14_3=0
if (cov(t(Y31),t(Y35))>0) cov15_3=cov(t(Y31),t(Y35)) else cov15_3=0
if (cov(t(Y31),t(Y36))>0) cov16_3=cov(t(Y31),t(Y36)) else cov16_3=0
if (cov(t(Y31),t(Y37))>0) cov17_3=cov(t(Y31),t(Y37)) else cov17_3=0
if (cov(t(Y32),t(Y33))>0) cov23_3=cov(t(Y32),t(Y33)) else cov23_3=0
if (cov(t(Y32),t(Y34))>0) cov24_3=cov(t(Y32),t(Y34)) else cov24_3=0
if (cov(t(Y32),t(Y35))>0) cov25_3=cov(t(Y32),t(Y35)) else cov25_3=0
if (cov(t(Y32),t(Y36))>0) cov26_3=cov(t(Y32),t(Y36)) else cov26_3=0
if (cov(t(Y32),t(Y37))>0) cov27_3=cov(t(Y32),t(Y37)) else cov27_3=0
if (cov(t(Y33),t(Y34))>0) cov34_3=cov(t(Y33),t(Y34)) else cov34_3=0
if (cov(t(Y33),t(Y35))>0) cov35_3=cov(t(Y33),t(Y35)) else cov35_3=0
if (cov(t(Y33),t(Y36))>0) cov36_3=cov(t(Y33),t(Y36)) else cov36_3=0
if (cov(t(Y33),t(Y37))>0) cov37_3=cov(t(Y33),t(Y37)) else cov37_3=0
if (cov(t(Y34),t(Y35))>0) cov45_3=cov(t(Y34),t(Y35)) else cov45_3=0
if (cov(t(Y34),t(Y36))>0) cov46_3=cov(t(Y34),t(Y36)) else cov46_3=0
if (cov(t(Y34),t(Y37))>0) cov47_3=cov(t(Y34),t(Y37)) else cov47_3=0
if (cov(t(Y35),t(Y36))>0) cov56_3=cov(t(Y35),t(Y36)) else cov56_3=0
if (cov(t(Y35),t(Y37))>0) cov57_3=cov(t(Y35),t(Y37)) else cov57_3=0
if (cov(t(Y36),t(Y37))>0) cov67_3=cov(t(Y36),t(Y37)) else cov67_3=0

cov3=c(cov12_3,cov13_3,cov14_3,cov15_3,cov16_3,cov17_3,cov23_3,cov24_3,
      cov25_3,cov26_3,cov27_3,cov34_3,cov35_3,cov36_3,cov37_3,cov45_3,
      cov46_3,cov47_3,cov56_3,cov57_3,cov67_3)

theta38=min(mean(cov3),(alfa3/(1+exp(-1*(delta3+gama3*t[1]))))+beta1,
            (alfa3/(1+exp(-1*(delta3+gama3*t[1]))))+beta2,
            (alfa3/(1+exp(-1*(delta3+gama3*t[1]))))+beta3,
            (alfa3/(1+exp(-1*(delta3+gama3*t[1]))))+beta4,
            (alfa3/(1+exp(-1*(delta3+gama3*t[1]))))+beta5,
            (alfa3/(1+exp(-1*(delta3+gama3*t[2]))))+beta1,
            (alfa3/(1+exp(-1*(delta3+gama3*t[2]))))+beta2,
            (alfa3/(1+exp(-1*(delta3+gama3*t[2]))))+beta3,
            (alfa3/(1+exp(-1*(delta3+gama3*t[2]))))+beta4,
            (alfa3/(1+exp(-1*(delta3+gama3*t[2]))))+beta5,
            (alfa3/(1+exp(-1*(delta3+gama3*t[3]))))+beta1,
            (alfa3/(1+exp(-1*(delta3+gama3*t[3]))))+beta2,
            (alfa3/(1+exp(-1*(delta3+gama3*t[3]))))+beta3,
            (alfa3/(1+exp(-1*(delta3+gama3*t[3]))))+beta4,
            (alfa3/(1+exp(-1*(delta3+gama3*t[3]))))+beta5,
            (alfa3/(1+exp(-1*(delta3+gama3*t[4]))))+beta1,
            (alfa3/(1+exp(-1*(delta3+gama3*t[4]))))+beta2,
            (alfa3/(1+exp(-1*(delta3+gama3*t[4]))))+beta3,
            (alfa3/(1+exp(-1*(delta3+gama3*t[4]))))+beta4,
            (alfa3/(1+exp(-1*(delta3+gama3*t[4]))))+beta5,

```

```

(alfa3/(1+exp(-1*(delta3+gama3*t[5]))))+beta1,
(alfa3/(1+exp(-1*(delta3+gama3*t[5]))))+beta2,
(alfa3/(1+exp(-1*(delta3+gama3*t[5]))))+beta3,
(alfa3/(1+exp(-1*(delta3+gama3*t[5]))))+beta4,
(alfa3/(1+exp(-1*(delta3+gama3*t[5]))))+beta5,
(alfa3/(1+exp(-1*(delta3+gama3*t[6]))))+beta1,
(alfa3/(1+exp(-1*(delta3+gama3*t[6]))))+beta2,
(alfa3/(1+exp(-1*(delta3+gama3*t[6]))))+beta3,
(alfa3/(1+exp(-1*(delta3+gama3*t[6]))))+beta4,
(alfa3/(1+exp(-1*(delta3+gama3*t[6]))))+beta5,
(alfa3/(1+exp(-1*(delta3+gama3*t[7]))))+beta1,
(alfa3/(1+exp(-1*(delta3+gama3*t[7]))))+beta2,
(alfa3/(1+exp(-1*(delta3+gama3*t[7]))))+beta3,
(alfa3/(1+exp(-1*(delta3+gama3*t[7]))))+beta4,
(alfa3/(1+exp(-1*(delta3+gama3*t[7]))))+beta5)
theta38_ini=theta38

if (cov(t(Y41),t(Y42))>0) cov12_4=cov(t(Y41),t(Y42)) else cov12_4=0
if (cov(t(Y41),t(Y43))>0) cov13_4=cov(t(Y41),t(Y43)) else cov13_4=0
if (cov(t(Y41),t(Y44))>0) cov14_4=cov(t(Y41),t(Y44)) else cov14_4=0
if (cov(t(Y41),t(Y45))>0) cov15_4=cov(t(Y41),t(Y45)) else cov15_4=0
if (cov(t(Y41),t(Y46))>0) cov16_4=cov(t(Y41),t(Y46)) else cov16_4=0
if (cov(t(Y41),t(Y47))>0) cov17_4=cov(t(Y41),t(Y47)) else cov17_4=0
if (cov(t(Y42),t(Y43))>0) cov23_4=cov(t(Y42),t(Y43)) else cov23_4=0
if (cov(t(Y42),t(Y44))>0) cov24_4=cov(t(Y42),t(Y44)) else cov24_4=0
if (cov(t(Y42),t(Y45))>0) cov25_4=cov(t(Y42),t(Y45)) else cov25_4=0
if (cov(t(Y42),t(Y46))>0) cov26_4=cov(t(Y42),t(Y46)) else cov26_4=0
if (cov(t(Y42),t(Y47))>0) cov27_4=cov(t(Y42),t(Y47)) else cov27_4=0
if (cov(t(Y43),t(Y44))>0) cov34_4=cov(t(Y43),t(Y44)) else cov34_4=0
if (cov(t(Y43),t(Y45))>0) cov35_4=cov(t(Y43),t(Y45)) else cov35_4=0
if (cov(t(Y43),t(Y46))>0) cov36_4=cov(t(Y43),t(Y46)) else cov36_4=0
if (cov(t(Y43),t(Y47))>0) cov37_4=cov(t(Y43),t(Y47)) else cov37_4=0
if (cov(t(Y44),t(Y45))>0) cov45_4=cov(t(Y44),t(Y45)) else cov45_4=0
if (cov(t(Y44),t(Y46))>0) cov46_4=cov(t(Y44),t(Y46)) else cov46_4=0
if (cov(t(Y44),t(Y47))>0) cov47_4=cov(t(Y44),t(Y47)) else cov47_4=0
if (cov(t(Y45),t(Y46))>0) cov56_4=cov(t(Y45),t(Y46)) else cov56_4=0
if (cov(t(Y45),t(Y47))>0) cov57_4=cov(t(Y45),t(Y47)) else cov57_4=0
if (cov(t(Y46),t(Y47))>0) cov67_4=cov(t(Y46),t(Y47)) else cov67_4=0

cov4=c(cov12_4,cov13_4,cov14_4,cov15_4,cov16_4,cov17_4,cov23_4,cov24_4,
      cov25_4,cov26_4,cov27_4,cov34_4,cov35_4,cov36_4,cov37_4,cov45_4,
      cov46_4,cov47_4,cov56_4,cov57_4,cov67_4)

theta48=min(mean(cov4),(alfa4/(1+exp(-1*(delta4+gama4*t[1]))))+beta1,
            (alfa4/(1+exp(-1*(delta4+gama4*t[1]))))+beta2,

```

```

(alfa4/(1+exp(-1*(delta4+gama4*t[1]))))+beta3,
(alfa4/(1+exp(-1*(delta4+gama4*t[1]))))+beta4,
(alfa4/(1+exp(-1*(delta4+gama4*t[1]))))+beta5,
(alfa4/(1+exp(-1*(delta4+gama4*t[2]))))+beta1,
(alfa4/(1+exp(-1*(delta4+gama4*t[2]))))+beta2,
(alfa4/(1+exp(-1*(delta4+gama4*t[2]))))+beta3,
(alfa4/(1+exp(-1*(delta4+gama4*t[2]))))+beta4,
(alfa4/(1+exp(-1*(delta4+gama4*t[2]))))+beta5,
(alfa4/(1+exp(-1*(delta4+gama4*t[3]))))+beta1,
(alfa4/(1+exp(-1*(delta4+gama4*t[3]))))+beta2,
(alfa4/(1+exp(-1*(delta4+gama4*t[3]))))+beta3,
(alfa4/(1+exp(-1*(delta4+gama4*t[3]))))+beta4,
(alfa4/(1+exp(-1*(delta4+gama4*t[3]))))+beta5,
(alfa4/(1+exp(-1*(delta4+gama4*t[4]))))+beta1,
(alfa4/(1+exp(-1*(delta4+gama4*t[4]))))+beta2,
(alfa4/(1+exp(-1*(delta4+gama4*t[4]))))+beta3,
(alfa4/(1+exp(-1*(delta4+gama4*t[4]))))+beta4,
(alfa4/(1+exp(-1*(delta4+gama4*t[4]))))+beta5,
(alfa4/(1+exp(-1*(delta4+gama4*t[5]))))+beta1,
(alfa4/(1+exp(-1*(delta4+gama4*t[5]))))+beta2,
(alfa4/(1+exp(-1*(delta4+gama4*t[5]))))+beta3,
(alfa4/(1+exp(-1*(delta4+gama4*t[5]))))+beta4,
(alfa4/(1+exp(-1*(delta4+gama4*t[5]))))+beta5,
(alfa4/(1+exp(-1*(delta4+gama4*t[6]))))+beta1,
(alfa4/(1+exp(-1*(delta4+gama4*t[6]))))+beta2,
(alfa4/(1+exp(-1*(delta4+gama4*t[6]))))+beta3,
(alfa4/(1+exp(-1*(delta4+gama4*t[6]))))+beta4,
(alfa4/(1+exp(-1*(delta4+gama4*t[6]))))+beta5,
(alfa4/(1+exp(-1*(delta4+gama4*t[7]))))+beta1,
(alfa4/(1+exp(-1*(delta4+gama4*t[7]))))+beta2,
(alfa4/(1+exp(-1*(delta4+gama4*t[7]))))+beta3,
(alfa4/(1+exp(-1*(delta4+gama4*t[7]))))+beta4,
(alfa4/(1+exp(-1*(delta4+gama4*t[7]))))+beta5)
theta48_ini=theta48

#####
#####Iniciando o processo iterativo#####
#####
nk=length(Yt1)
m2lnL=1000; m2lnlike=1000
m2lnL1=0; m2lnL2=0
iter=0
difllike1=1000; difllike2=1000
prec=1e-6; maxit=500

```

```

while ((difllike1 > prec) & (iter < maxit)) {
  iter=iter+1
  m2lnL1=m2lnL; m2lnL2=m2lnlike

  #####Iniciando o Passo E do Algoritmo EM#####/
  numer8=matrix((rep(0,nk)),nrow=nk,ncol=1)
  denom=matrix((rep(0,nk)),nrow=nk,ncol=1)
  s1=matrix((rep(0,nk)),nrow=nk,ncol=1); s2=matrix((rep(0,nk)),nrow=nk,ncol=1)
  s3=matrix((rep(0,nk)),nrow=nk,ncol=1); s4=matrix((rep(0,nk)),nrow=nk,ncol=1)
  s5=matrix((rep(0,nk)),nrow=nk,ncol=1); s6=matrix((rep(0,nk)),nrow=nk,ncol=1)
  s7=matrix((rep(0,nk)),nrow=nk,ncol=1); s8=matrix((rep(0,nk)),nrow=nk,ncol=1)
  lnlike=matrix((rep(0,nk)),nrow=nk,ncol=1)

  #### Encontrando as esperanças das variáveis latentes #####
  for (i in 1:nk){
    for (w8 in 0:min(Yt1[i],Yt2[i],Yt3[i],Yt4[i],Yt5[i],Yt6[i],Yt7[i]))){
      lp=(-1*theta18)+log((theta18+0.00001)**w8)+
        (-1*((alfa1/(1+exp(-1*(delta1+gama1*t[1]))))+
          beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta18))+
        (-1*((alfa1/(1+exp(-1*(delta1+gama1*t[2]))))+
          beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta18))+
        (-1*((alfa1/(1+exp(-1*(delta1+gama1*t[3]))))+
          beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta18))+
        (-1*((alfa1/(1+exp(-1*(delta1+gama1*t[4]))))+
          beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta18))+
        (-1*((alfa1/(1+exp(-1*(delta1+gama1*t[5]))))+
          beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta18))+
        (-1*((alfa1/(1+exp(-1*(delta1+gama1*t[6]))))+
          beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta18))+
        (-1*((alfa1/(1+exp(-1*(delta1+gama1*t[7]))))+
          beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta18))+
        (Yt1[i]-w8)*log((((alfa1/(1+exp(-1*(delta1+gama1*t[1]))))+
          beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta18+0.00001))))+
        (Yt2[i]-w8)*log((((alfa1/(1+exp(-1*(delta1+gama1*t[2]))))+
          beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta18+0.00001))))+
        (Yt3[i]-w8)*log((((alfa1/(1+exp(-1*(delta1+gama1*t[3]))))+
          beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta18+0.00001))))+
        (Yt4[i]-w8)*log((((alfa1/(1+exp(-1*(delta1+gama1*t[4]))))+
          beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta18+0.00001))))+
        (Yt5[i]-w8)*log((((alfa1/(1+exp(-1*(delta1+gama1*t[5]))))+
          beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta18+0.00001))))+
        (Yt6[i]-w8)*log((((alfa1/(1+exp(-1*(delta1+gama1*t[6]))))+
          beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta18+0.00001))))+
        (Yt7[i]-w8)*log((((alfa1/(1+exp(-1*(delta1+gama1*t[7]))))+
          beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta18+0.00001))))-

```

```

lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-
lgamma(Yt3[i]-w8+1)-lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-
lgamma(Yt6[i]-w8+1)-lgamma(Yt7[i]-w8+1))*trat1[i]+
((-1*theta28)+log((theta28+0.00001)**w8)+
(-1*((alfa2/(1+exp(-1*(delta2+gama2*t[1]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta28))+
(-1*((alfa2/(1+exp(-1*(delta2+gama2*t[2]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta28))+
(-1*((alfa2/(1+exp(-1*(delta2+gama2*t[3]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta28))+
(-1*((alfa2/(1+exp(-1*(delta2+gama2*t[4]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta28))+
(-1*((alfa2/(1+exp(-1*(delta2+gama2*t[5]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta28))+
(-1*((alfa2/(1+exp(-1*(delta2+gama2*t[6]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta28))+
(-1*((alfa2/(1+exp(-1*(delta2+gama2*t[7]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta28))+
(Yt1[i]-w8)*log((((alfa2/(1+exp(-1*(delta2+gama2*t[1]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta28+0.00001))))+
(Yt2[i]-w8)*log((((alfa2/(1+exp(-1*(delta2+gama2*t[2]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta28+0.00001))))+
(Yt3[i]-w8)*log((((alfa2/(1+exp(-1*(delta2+gama2*t[3]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta28+0.00001))))+
(Yt4[i]-w8)*log((((alfa2/(1+exp(-1*(delta2+gama2*t[4]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta28+0.00001))))+
(Yt5[i]-w8)*log((((alfa2/(1+exp(-1*(delta2+gama2*t[5]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta28+0.00001))))+
(Yt6[i]-w8)*log((((alfa2/(1+exp(-1*(delta2+gama2*t[6]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta28+0.00001))))+
(Yt7[i]-w8)*log((((alfa2/(1+exp(-1*(delta2+gama2*t[7]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta28+0.00001))))-
lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-
lgamma(Yt3[i]-w8+1)-lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-
lgamma(Yt6[i]-w8+1)-lgamma(Yt7[i]-w8+1))*trat2[i]+
((-1*theta38)+log((theta38+0.00001)**w8)+
(-1*((alfa3/(1+exp(-1*(delta3+gama3*t[1]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta38))+
(-1*((alfa3/(1+exp(-1*(delta3+gama3*t[2]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta38))+
(-1*((alfa3/(1+exp(-1*(delta3+gama3*t[3]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta38))+
(-1*((alfa3/(1+exp(-1*(delta3+gama3*t[4]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta38))+
(-1*((alfa3/(1+exp(-1*(delta3+gama3*t[5]))))+

```

```

beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta38))+
  (-1*((alfa3/(1+exp(-1*(delta3+gama3*t[6]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta38))+
  (-1*((alfa3/(1+exp(-1*(delta3+gama3*t[7]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta38))+
  (Yt1[i]-w8)*log((((alfa3/(1+exp(-1*(delta3+gama3*t[1]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta38+0.00001))))+
  (Yt2[i]-w8)*log((((alfa3/(1+exp(-1*(delta3+gama3*t[2]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta38+0.00001))))+
  (Yt3[i]-w8)*log((((alfa3/(1+exp(-1*(delta3+gama3*t[3]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta38+0.00001))))+
  (Yt4[i]-w8)*log((((alfa3/(1+exp(-1*(delta3+gama3*t[4]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta38+0.00001))))+
  (Yt5[i]-w8)*log((((alfa3/(1+exp(-1*(delta3+gama3*t[5]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta38+0.00001))))+
  (Yt6[i]-w8)*log((((alfa3/(1+exp(-1*(delta3+gama3*t[6]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta38+0.00001))))+
  (Yt7[i]-w8)*log((((alfa3/(1+exp(-1*(delta3+gama3*t[7]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta38+0.00001))))-
  lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-
  lgamma(Yt3[i]-w8+1)-lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-
  lgamma(Yt6[i]-w8+1)-lgamma(Yt7[i]-w8+1))*trat3[i]+
  ((-1*theta48)+log((theta48+0.00001)**w8)+
  (-1*((alfa4/(1+exp(-1*(delta4+gama4*t[1]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta48))+
  (-1*((alfa4/(1+exp(-1*(delta4+gama4*t[2]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta48))+
  (-1*((alfa4/(1+exp(-1*(delta4+gama4*t[3]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta48))+
  (-1*((alfa4/(1+exp(-1*(delta4+gama4*t[4]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta48))+
  (-1*((alfa4/(1+exp(-1*(delta4+gama4*t[5]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta48))+
  (-1*((alfa4/(1+exp(-1*(delta4+gama4*t[6]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta48))+
  (-1*((alfa4/(1+exp(-1*(delta4+gama4*t[7]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta48))+
  (Yt1[i]-w8)*log((((alfa4/(1+exp(-1*(delta4+gama4*t[1]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta48+0.00001))))+
  (Yt2[i]-w8)*log((((alfa4/(1+exp(-1*(delta4+gama4*t[2]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta48+0.00001))))+
  (Yt3[i]-w8)*log((((alfa4/(1+exp(-1*(delta4+gama4*t[3]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta48+0.00001))))+
  (Yt4[i]-w8)*log((((alfa4/(1+exp(-1*(delta4+gama4*t[4]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta48+0.00001))))+
  (Yt5[i]-w8)*log((((alfa4/(1+exp(-1*(delta4+gama4*t[5]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta48+0.00001))))+
  (Yt6[i]-w8)*log((((alfa4/(1+exp(-1*(delta4+gama4*t[6]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta48+0.00001))))+
  (Yt7[i]-w8)*log((((alfa4/(1+exp(-1*(delta4+gama4*t[7]))))+
beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta48+0.00001))))+

```



```

      (Yt5[i]-w8)*log((((alfa4/(1+exp(-1*(delta4+gama4*t[5]))))+
      beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta48+0.00001)))+
      (Yt6[i]-w8)*log((((alfa4/(1+exp(-1*(delta4+gama4*t[6]))))+
      beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta48+0.00001)))+
      (Yt7[i]-w8)*log((((alfa4/(1+exp(-1*(delta4+gama4*t[7]))))+
      beta2*b2[i]+beta3*b3[i]+beta4*b4[i]+beta5*b5[i]-theta48+0.00001)))-
      lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-
      lgamma(Yt3[i]-w8+1)-lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-
      lgamma(Yt6[i]-w8+1)-lgamma(Yt7[i]-w8+1))*trat4[i]

      p=exp(lp)
      prod=w8*p
      numer8[i]=numer8[i]+prod
      denom[i]=denom[i]+p
    }
    s8[i]=numer8[i]/denom[i]
    s1[i]=Yt1[i]-s8[i]; s2[i]=Yt2[i]-s8[i]; s3[i]=Yt3[i]-s8[i]; s4[i]=Yt4[i]-s8[i];
    s5[i]=Yt5[i]-s8[i]; s6[i]=Yt6[i]-s8[i]; s7[i]=Yt7[i]-s8[i]
    lnlike[i]=log(denom[i])
  }
  #####Fim do Passo E#####

  #####Início do Passo M#####
  options(warn = 0)
  out1<-optim(c(alfa1,delta1,gama1,theta18,alfa2,delta2,gama2,theta28,
               alfa3,delta3,gama3,theta38,alfa4,delta4,gama4,theta48,
               beta2,beta3,beta4,beta5),fn,sgrd,method="BFGS",
             control=list(maxit=1000),hessian=TRUE)

  alfa1=out1$par[1];delta1=out1$par[2];gama1=out1$par[3];theta18=out1$par[4]
  alfa2=out1$par[5];delta2=out1$par[6];gama2=out1$par[7];theta28=out1$par[8]
  alfa3=out1$par[9];delta3=out1$par[10];gama3=out1$par[11];theta38=out1$par[12]
  alfa4=out1$par[13];delta4=out1$par[14];gama4=out1$par[15];theta48=out1$par[16]

  beta1=0; beta2=out1$par[17]; beta3=out1$par[18]; beta4=out1$par[19];
  beta5=out1$par[20]

  converge=out1$convergence; names(converge)<-('converge')

  m2lnL=2*out1$value; m2lnlike=-2*sum(lnlike)
  difllike1=abs((m2lnL-m2lnL1)/m2lnL1); difllike2=abs((m2lnlike-m2lnL2)/m2lnL2)

  names(nsimul) <- ('Simul'); names(iter)<-('iter')
  printvectort1<-c(alfa1,delta1,gama1,theta18)
  printvectort2<-c(alfa2,delta2,gama2,theta28)

```

```

printvectort3<-c(alfa3,delta3,gama3,theta38)
printvectort4<-c(alfa4,delta4,gama4,theta48)
printvectorb<-c(beta1,beta2,beta3,beta4,beta5)

printvectorvero<-c(m2lnL,difllike1,m2lnlike,difllike2)

names(printvectort1)<-c('alfa1','delta1','gama1','theta18')
names(printvectort2)<-c('alfa2','delta2','gama2','theta28')
names(printvectort3)<-c('alfa3','delta3','gama3','theta38')
names(printvectort4)<-c('alfa4','delta4','gama4','theta48')
names(printvectorb)<-c('beta1','beta2','beta3','beta4','beta5')

names(printvectorvero)<-c('-2*logvero1','Rel.Dif.logvero1',
                          '-2*logvero2','Rel.Dif.logvero2')

print.default(nsimul); print.default(iter)
print.default(converge,digits=10)
print.default(printvectort1,digits=10)
print.default(printvectort2,digits=10)
print.default(printvectort3,digits=10)
print.default(printvectort4,digits=10)
print.default(printvectorb,digits=10)
print.default(printvectorvero,digits=10)
}
result[k,]=c(nsimul,iter,converge,m2lnL,difllike1,m2lnlike,difllike2,
             alfa1_ini,delta1_ini,gama1_ini,theta18_ini,
             alfa2_ini,delta2_ini,gama2_ini,theta28_ini,
             alfa3_ini,delta3_ini,gama3_ini,theta38_ini,
             alfa4_ini,delta4_ini,gama4_ini,theta48_ini,
             beta1_ini,beta2_ini,beta3_ini,beta4_ini,beta5_ini)
result[k+1,]=c(nsimul,iter,converge,m2lnL,difllike1,m2lnlike,difllike2,
              alfa1,delta1,gama1,theta18,alfa2,delta2,gama2,theta28,
              alfa3,delta3,gama3,theta38,alfa4,delta4,gama4,theta48,
              beta1,beta2,beta3,beta4,beta5)

k=k+2
nsimul=nsimul+1
}

```

B.3 - Delineamento em blocos casualizados com efeito aleatório para blocos

```

library(nlreg)

# PARÂMETROS DA CURVA LOGÍSTICA PARA O TRATAMENTO 1
alfa1p=359.916; delta1p=-3.769; gama1p=0.022
# PARÂMETROS DA CURVA LOGÍSTICA PARA O TRATAMENTO 2
alfa2p=391.870; delta2p=-3.441; gama2p=0.021

```



```

# PARÂMETROS DA CURVA LOGÍSTICA PARA O TRATAMENTO 3
alfa3p=374.896; delta3p=-3.536; gama3p=0.022
# PARÂMETROS DA CURVA LOGÍSTICA PARA O TRATAMENTO 4
alfa4p=334.128; delta4p=-3.981; gama4p=0.026

#Cálculos dos valores sobre a curva em cada tempo
t=c(1,46,91,136,181,226,271)

media = function(alfa,delta,gama,t){
  mu=matrix(numeric(length(t)),ncol=1)
  for (i in 1:length(t)){
    mu[i]=(alfa/(1+exp(-(delta+gama*t[i]))))
  }
  mu
}

mu1=media(alfa1p,delta1p,gama1p,t); mu2=media(alfa2p,delta2p,gama2p,t)
mu3=media(alfa3p,delta3p,gama3p,t); mu4=media(alfa4p,delta4p,gama4p,t)

##### Parâmetros de Covariância #####
theta18p=8; theta28p=12; theta38p=10; theta48p=6

##### Variação dentro de blocos #
sigma2bp=79.224

n=5
trat=c(rep(1,n),rep(2,n),rep(3,n),rep(4,n))
trat1=c(rep(1,n),rep(0,3*n))
trat2=c(rep(0,n),rep(1,n),rep(0,2*n))
trat3=c(rep(0,2*n),rep(1,n),rep(0,n))
trat4=c(rep(0,3*n),rep(1,n))
bloco=rep(1:n,4)

#### Cálculo da esperança condicional de bi/Yi #####
febdy <- function(p) {
  num=matrix(0,nrow=n,ncol=1); den=matrix(0,nrow=n,ncol=1);
  ebdy=matrix(0,nrow=n,ncol=1)
  for (i in 1:n){
    s18=0; s28=0; s38=0; s48=0
    for (w18 in 0:min(Y11[i],Y12[i],Y13[i],Y14[i],Y15[i],Y16[i],Y17[i])){
      ls18=(-1*p[4])+log((p[4])**w18)+
        (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[1]))))+xk*p[17]-p[4]))+
        (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[2]))))+xk*p[17]-p[4]))+
        (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[3]))))+xk*p[17]-p[4]))+
        (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[4]))))+xk*p[17]-p[4]))+
        (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[5]))))+xk*p[17]-p[4]))+

```

```

(-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[6]))))+xk*p[17]-p[4]))+
(-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[7]))))+xk*p[17]-p[4]))+
(Y11[i]-w18)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[1]))))+xk*p[17]-p[4]))+
(Y12[i]-w18)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[2]))))+xk*p[17]-p[4]))+
(Y13[i]-w18)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[3]))))+xk*p[17]-p[4]))+
(Y14[i]-w18)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[4]))))+xk*p[17]-p[4]))+
(Y15[i]-w18)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[5]))))+xk*p[17]-p[4]))+
(Y16[i]-w18)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[6]))))+xk*p[17]-p[4]))+
(Y17[i]-w18)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[7]))))+xk*p[17]-p[4]))-
lgamma(w18+1)-lgamma(Y11[i]-w18+1)-lgamma(Y12[i]-w18+1)-
lgamma(Y13[i]-w18+1)-lgamma(Y14[i]-w18+1)-lgamma(Y15[i]-w18+1)-
lgamma(Y16[i]-w18+1)-lgamma(Y17[i]-w18+1)
s18=s18+exp(ls18)
}
for (w28 in 0:min(Y21[i],Y22[i],Y23[i],Y24[i],Y25[i],Y26[i],Y27[i])){
ls28=(-1*p[8])+log((p[8])**w28)+
(-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[1]))))+xk*p[17]-p[8]))+
(-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[2]))))+xk*p[17]-p[8]))+
(-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[3]))))+xk*p[17]-p[8]))+
(-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[4]))))+xk*p[17]-p[8]))+
(-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[5]))))+xk*p[17]-p[8]))+
(-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[6]))))+xk*p[17]-p[8]))+
(-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[7]))))+xk*p[17]-p[8]))+
(Y21[i]-w28)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[1]))))+xk*p[17]-p[8]))+
(Y22[i]-w28)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[2]))))+xk*p[17]-p[8]))+
(Y23[i]-w28)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[3]))))+xk*p[17]-p[8]))+
(Y24[i]-w28)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[4]))))+xk*p[17]-p[8]))+
(Y25[i]-w28)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[5]))))+xk*p[17]-p[8]))+
(Y26[i]-w28)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[6]))))+xk*p[17]-p[8]))+
(Y27[i]-w28)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[7]))))+xk*p[17]-p[8]))-
lgamma(w28+1)-lgamma(Y21[i]-w28+1)-lgamma(Y22[i]-w28+1)-
lgamma(Y23[i]-w28+1)-lgamma(Y24[i]-w28+1)-lgamma(Y25[i]-w28+1)-
lgamma(Y26[i]-w28+1)-lgamma(Y27[i]-w28+1)
s28=s28+exp(ls28)
}
for (w38 in 0:min(Y31[i],Y32[i],Y33[i],Y34[i],Y35[i],Y36[i],Y37[i])){
ls38=(-1*p[12])+log((p[12])**w38)+
(-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[1]))))+xk*p[17]-p[12]))+
(-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[2]))))+xk*p[17]-p[12]))+
(-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[3]))))+xk*p[17]-p[12]))+
(-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[4]))))+xk*p[17]-p[12]))+
(-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[5]))))+xk*p[17]-p[12]))+
(-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[6]))))+xk*p[17]-p[12]))+
(-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[7]))))+xk*p[17]-p[12]))+
(Y31[i]-w38)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[1]))))+xk*p[17]-p[12]))+

```

```

(Y32[i]-w38)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[2]))))+xk*p[17]-p[12])))+
(Y33[i]-w38)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[3]))))+xk*p[17]-p[12])))+
(Y34[i]-w38)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[4]))))+xk*p[17]-p[12])))+
(Y35[i]-w38)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[5]))))+xk*p[17]-p[12])))+
(Y36[i]-w38)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[6]))))+xk*p[17]-p[12])))+
(Y37[i]-w38)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[7]))))+xk*p[17]-p[12])))-
lgamma(w38+1)-lgamma(Y31[i]-w38+1)-lgamma(Y32[i]-w38+1)-
lgamma(Y33[i]-w38+1)-lgamma(Y34[i]-w38+1)-lgamma(Y35[i]-w38+1)-
lgamma(Y36[i]-w38+1)-lgamma(Y37[i]-w38+1)
    s38=s38+exp(ls38)
}
for (w48 in 0:min(Y41[i],Y42[i],Y43[i],Y44[i],Y45[i],Y46[i],Y47[i]))){
ls48=(-1*p[16])+log((p[16])**w48)+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[1]))))+xk*p[17]-p[16]))+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[2]))))+xk*p[17]-p[16]))+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[3]))))+xk*p[17]-p[16]))+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[4]))))+xk*p[17]-p[16]))+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[5]))))+xk*p[17]-p[16]))+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[6]))))+xk*p[17]-p[16]))+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[7]))))+xk*p[17]-p[16]))+
(Y41[i]-w48)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[1]))))+xk*p[17]-p[16])))+
(Y42[i]-w48)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[2]))))+xk*p[17]-p[16])))+
(Y43[i]-w48)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[3]))))+xk*p[17]-p[16])))+
(Y44[i]-w48)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[4]))))+xk*p[17]-p[16])))+
(Y45[i]-w48)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[5]))))+xk*p[17]-p[16])))+
(Y46[i]-w48)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[6]))))+xk*p[17]-p[16])))+
(Y47[i]-w48)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[7]))))+xk*p[17]-p[16])))-
lgamma(w48+1)-lgamma(Y41[i]-w48+1)-lgamma(Y42[i]-w48+1)-
lgamma(Y43[i]-w48+1)-lgamma(Y44[i]-w48+1)-lgamma(Y45[i]-w48+1)-
lgamma(Y46[i]-w48+1)-lgamma(Y47[i]-w48+1)
    s48=s48+exp(ls48)
}
fxknum=(xk*p[17])*s18*s28*s38*s48*(1/gamma(1/p[17]))*((1/p[17])**((1/p[17]))*
((xk*p[17])**((1/p[17])-1))*p[17]
Akfxknum=Ak*fxknum
num[i]=sum(Akfxknum)
fxkden=s18*s28*s38*s48*(1/gamma(1/p[17]))*((1/p[17])**((1/p[17]))*
((xk*p[17])**((1/p[17])-1))*p[17]
Akfxkden=Ak*fxkden
den[i]=sum(Akfxkden)
ebdy[i]=num[i]/den[i]
}
ebdy
}

```

```

#### Cálculo do log(distribuição) marginal de Yij #####
lfy <- function(p) {
  fy=matrix(0,nrow=n,ncol=1)
  logfy=matrix(0,nrow=n,ncol=1)
  for (i in 1:n){
    s18=0; s28=0; s38=0; s48=0
    for (w18 in 0:min(Y11[i],Y12[i],Y13[i],Y14[i],Y15[i],Y16[i],Y17[i])){
      ls18=(-1*p[4])+log((p[4])**w18)+
        (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[1]))))+xk*p[17]-p[4]))+
        (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[2]))))+xk*p[17]-p[4]))+
        (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[3]))))+xk*p[17]-p[4]))+
        (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[4]))))+xk*p[17]-p[4]))+
        (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[5]))))+xk*p[17]-p[4]))+
        (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[6]))))+xk*p[17]-p[4]))+
        (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[7]))))+xk*p[17]-p[4]))+
        (Y11[i]-w18)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[1]))))+xk*p[17]-p[4]))+
        (Y12[i]-w18)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[2]))))+xk*p[17]-p[4]))+
        (Y13[i]-w18)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[3]))))+xk*p[17]-p[4]))+
        (Y14[i]-w18)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[4]))))+xk*p[17]-p[4]))+
        (Y15[i]-w18)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[5]))))+xk*p[17]-p[4]))+
        (Y16[i]-w18)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[6]))))+xk*p[17]-p[4]))+
        (Y17[i]-w18)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[7]))))+xk*p[17]-p[4]))-
        lgamma(w18+1)-lgamma(Y11[i]-w18+1)-lgamma(Y12[i]-w18+1)-
        lgamma(Y13[i]-w18+1)-lgamma(Y14[i]-w18+1)-lgamma(Y15[i]-w18+1)-
        lgamma(Y16[i]-w18+1)-lgamma(Y17[i]-w18+1)
      s18=s18+exp(ls18)
    }
    for (w28 in 0:min(Y21[i],Y22[i],Y23[i],Y24[i],Y25[i],Y26[i],Y27[i])){
      ls28=(-1*p[8])+log((p[8])**w28)+
        (-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[1]))))+xk*p[17]-p[8]))+
        (-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[2]))))+xk*p[17]-p[8]))+
        (-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[3]))))+xk*p[17]-p[8]))+
        (-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[4]))))+xk*p[17]-p[8]))+
        (-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[5]))))+xk*p[17]-p[8]))+
        (-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[6]))))+xk*p[17]-p[8]))+
        (-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[7]))))+xk*p[17]-p[8]))+
        (Y21[i]-w28)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[1]))))+xk*p[17]-p[8]))+
        (Y22[i]-w28)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[2]))))+xk*p[17]-p[8]))+
        (Y23[i]-w28)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[3]))))+xk*p[17]-p[8]))+
        (Y24[i]-w28)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[4]))))+xk*p[17]-p[8]))+
        (Y25[i]-w28)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[5]))))+xk*p[17]-p[8]))+
        (Y26[i]-w28)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[6]))))+xk*p[17]-p[8]))+
        (Y27[i]-w28)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[7]))))+xk*p[17]-p[8]))-
        lgamma(w28+1)-lgamma(Y21[i]-w28+1)-lgamma(Y22[i]-w28+1)-
        lgamma(Y23[i]-w28+1)-lgamma(Y24[i]-w28+1)-lgamma(Y25[i]-w28+1)-

```

```

lgamma(Y26[i]-w28+1)-lgamma(Y27[i]-w28+1)
s28=s28+exp(ls28)
}
for (w38 in 0:min(Y31[i],Y32[i],Y33[i],Y34[i],Y35[i],Y36[i],Y37[i])){
ls38=(-1*p[12])+log((p[12])**w38)+
(-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[1]))))+xk*p[17]-p[12]))+
(-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[2]))))+xk*p[17]-p[12]))+
(-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[3]))))+xk*p[17]-p[12]))+
(-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[4]))))+xk*p[17]-p[12]))+
(-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[5]))))+xk*p[17]-p[12]))+
(-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[6]))))+xk*p[17]-p[12]))+
(-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[7]))))+xk*p[17]-p[12]))+
(Y31[i]-w38)*log((((p[9]/(1+exp(-1*(p[10]+p[11]*t[1]))))+xk*p[17]-p[12])))+
(Y32[i]-w38)*log((((p[9]/(1+exp(-1*(p[10]+p[11]*t[2]))))+xk*p[17]-p[12])))+
(Y33[i]-w38)*log((((p[9]/(1+exp(-1*(p[10]+p[11]*t[3]))))+xk*p[17]-p[12])))+
(Y34[i]-w38)*log((((p[9]/(1+exp(-1*(p[10]+p[11]*t[4]))))+xk*p[17]-p[12])))+
(Y35[i]-w38)*log((((p[9]/(1+exp(-1*(p[10]+p[11]*t[5]))))+xk*p[17]-p[12])))+
(Y36[i]-w38)*log((((p[9]/(1+exp(-1*(p[10]+p[11]*t[6]))))+xk*p[17]-p[12])))+
(Y37[i]-w38)*log((((p[9]/(1+exp(-1*(p[10]+p[11]*t[7]))))+xk*p[17]-p[12])))-
lgamma(w38+1)-lgamma(Y31[i]-w38+1)-lgamma(Y32[i]-w38+1)-
lgamma(Y33[i]-w38+1)-lgamma(Y34[i]-w38+1)-lgamma(Y35[i]-w38+1)-
lgamma(Y36[i]-w38+1)-lgamma(Y37[i]-w38+1)
s38=s38+exp(ls38)
}
for (w48 in 0:min(Y41[i],Y42[i],Y43[i],Y44[i],Y45[i],Y46[i],Y47[i])){
ls48=(-1*p[16])+log((p[16])**w48)+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[1]))))+xk*p[17]-p[16]))+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[2]))))+xk*p[17]-p[16]))+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[3]))))+xk*p[17]-p[16]))+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[4]))))+xk*p[17]-p[16]))+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[5]))))+xk*p[17]-p[16]))+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[6]))))+xk*p[17]-p[16]))+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[7]))))+xk*p[17]-p[16]))+
(Y41[i]-w48)*log((((p[13]/(1+exp(-1*(p[14]+p[15]*t[1]))))+xk*p[17]-p[16])))+
(Y42[i]-w48)*log((((p[13]/(1+exp(-1*(p[14]+p[15]*t[2]))))+xk*p[17]-p[16])))+
(Y43[i]-w48)*log((((p[13]/(1+exp(-1*(p[14]+p[15]*t[3]))))+xk*p[17]-p[16])))+
(Y44[i]-w48)*log((((p[13]/(1+exp(-1*(p[14]+p[15]*t[4]))))+xk*p[17]-p[16])))+
(Y45[i]-w48)*log((((p[13]/(1+exp(-1*(p[14]+p[15]*t[5]))))+xk*p[17]-p[16])))+
(Y46[i]-w48)*log((((p[13]/(1+exp(-1*(p[14]+p[15]*t[6]))))+xk*p[17]-p[16])))+
(Y47[i]-w48)*log((((p[13]/(1+exp(-1*(p[14]+p[15]*t[7]))))+xk*p[17]-p[16])))-
lgamma(w48+1)-lgamma(Y41[i]-w48+1)-lgamma(Y42[i]-w48+1)-
lgamma(Y43[i]-w48+1)-lgamma(Y44[i]-w48+1)-lgamma(Y45[i]-w48+1)-
lgamma(Y46[i]-w48+1)-lgamma(Y47[i]-w48+1)
s48=s48+exp(ls48)
}

```

```

fxk=s18*s28*s38*s48*(1/gamma(1/p[17]))*((1/p[17])**((1/p[17]))*
  ((xk*p[17])**((1/p[17])-1))*p[17]
Akfxk=Ak*fxk
fy[i]=sum(Akfxk)
logfy[i]=log(fy[i])
  }
lnfy=sum(logfy)
lnfy
}

### Cálculo da esperança condicional de sij8=(Wij8/b_i,Yij) #####
few8dby <- function(p) {
  s18=matrix(0,nrow=n,ncol=1); s28=matrix(0,nrow=n,ncol=1)
  s38=matrix(0,nrow=n,ncol=1); s48=matrix(0,nrow=n,ncol=1)
  for (i in 1:n){
    s18num=0; s18den=0; s28num=0; s28den=0; s38num=0; s38den=0;
    s48num=0; s48den=0
    for (w18 in 0:min(Y11[i],Y12[i],Y13[i],Y14[i],Y15[i],Y16[i],Y17[i])){
      ls18=(-1*p[4])+log((p[4])**w18)+
        (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[1]))))+ebdy[i]-p[4]))+
        (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[2]))))+ebdy[i]-p[4]))+
        (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[3]))))+ebdy[i]-p[4]))+
        (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[4]))))+ebdy[i]-p[4]))+
        (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[5]))))+ebdy[i]-p[4]))+
        (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[6]))))+ebdy[i]-p[4]))+
        (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[7]))))+ebdy[i]-p[4]))+
        (Y11[i]-w18)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[1]))))+ebdy[i]-p[4]))+
        (Y12[i]-w18)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[2]))))+ebdy[i]-p[4]))+
        (Y13[i]-w18)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[3]))))+ebdy[i]-p[4]))+
        (Y14[i]-w18)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[4]))))+ebdy[i]-p[4]))+
        (Y15[i]-w18)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[5]))))+ebdy[i]-p[4]))+
        (Y16[i]-w18)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[6]))))+ebdy[i]-p[4]))+
        (Y17[i]-w18)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[7]))))+ebdy[i]-p[4]))-
        lgamma(w18+1)-lgamma(Y11[i]-w18+1)-lgamma(Y12[i]-w18+1)-
        lgamma(Y13[i]-w18+1)-lgamma(Y14[i]-w18+1)-lgamma(Y15[i]-w18+1)-
        lgamma(Y16[i]-w18+1)-lgamma(Y17[i]-w18+1)
      s18num=s18num+w18*exp(ls18)
      s18den=s18den+exp(ls18)
    }
    s18[i]=s18num/s18den

    for (w28 in 0:min(Y21[i],Y22[i],Y23[i],Y24[i],Y25[i],Y26[i],Y27[i])){
      ls28=(-1*p[8])+log((p[8])**w28)+
        (-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[1]))))+ebdy[i]-p[8]))+
        (-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[2]))))+ebdy[i]-p[8]))+

```



```

(-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[3]))))+ebdy[i]-p[8]))+
(-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[4]))))+ebdy[i]-p[8]))+
(-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[5]))))+ebdy[i]-p[8]))+
(-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[6]))))+ebdy[i]-p[8]))+
(-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[7]))))+ebdy[i]-p[8]))+
(Y21[i]-w28)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[1]))))+ebdy[i]-p[8]))+
(Y22[i]-w28)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[2]))))+ebdy[i]-p[8]))+
(Y23[i]-w28)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[3]))))+ebdy[i]-p[8]))+
(Y24[i]-w28)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[4]))))+ebdy[i]-p[8]))+
(Y25[i]-w28)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[5]))))+ebdy[i]-p[8]))+
(Y26[i]-w28)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[6]))))+ebdy[i]-p[8]))+
(Y27[i]-w28)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[7]))))+ebdy[i]-p[8]))-
lgamma(w28+1)-lgamma(Y21[i]-w28+1)-lgamma(Y22[i]-w28+1)-
lgamma(Y23[i]-w28+1)-lgamma(Y24[i]-w28+1)-lgamma(Y25[i]-w28+1)-
lgamma(Y26[i]-w28+1)-lgamma(Y27[i]-w28+1)
    s28num=s28num+w28*exp(ls28)
    s28den=s28den+exp(ls28)
}
s28[i]=s28num/s28den

for (w38 in 0:min(Y31[i],Y32[i],Y33[i],Y34[i],Y35[i],Y36[i],Y37[i]))){
  ls38=(-1*p[12])+log((p[12])**w38)+
  (-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[1]))))+ebdy[i]-p[12]))+
  (-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[2]))))+ebdy[i]-p[12]))+
  (-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[3]))))+ebdy[i]-p[12]))+
  (-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[4]))))+ebdy[i]-p[12]))+
  (-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[5]))))+ebdy[i]-p[12]))+
  (-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[6]))))+ebdy[i]-p[12]))+
  (-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[7]))))+ebdy[i]-p[12]))+
  (Y31[i]-w38)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[1]))))+ebdy[i]-p[12]))+
  (Y32[i]-w38)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[2]))))+ebdy[i]-p[12]))+
  (Y33[i]-w38)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[3]))))+ebdy[i]-p[12]))+
  (Y34[i]-w38)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[4]))))+ebdy[i]-p[12]))+
  (Y35[i]-w38)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[5]))))+ebdy[i]-p[12]))+
  (Y36[i]-w38)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[6]))))+ebdy[i]-p[12]))+
  (Y37[i]-w38)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[7]))))+ebdy[i]-p[12]))-
  lgamma(w38+1)-lgamma(Y31[i]-w38+1)-lgamma(Y32[i]-w38+1)-
  lgamma(Y33[i]-w38+1)-lgamma(Y34[i]-w38+1)-lgamma(Y35[i]-w38+1)-
  lgamma(Y36[i]-w38+1)-lgamma(Y37[i]-w38+1)
    s38num=s38num+w38*exp(ls38)
    s38den=s38den+exp(ls38)
}
s38[i]=s38num/s38den

for (w48 in 0:min(Y41[i],Y42[i],Y43[i],Y44[i],Y45[i],Y46[i],Y47[i])){

```

```

ls48=(-1*p[16])+log((p[16])**w48)+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[1]))))+ebdy[i]-p[16]))+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[2]))))+ebdy[i]-p[16]))+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[3]))))+ebdy[i]-p[16]))+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[4]))))+ebdy[i]-p[16]))+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[5]))))+ebdy[i]-p[16]))+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[6]))))+ebdy[i]-p[16]))+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[7]))))+ebdy[i]-p[16]))+
(Y41[i]-w48)*log((((p[13]/(1+exp(-1*(p[14]+p[15]*t[1]))))+ebdy[i]-p[16])))+
(Y42[i]-w48)*log((((p[13]/(1+exp(-1*(p[14]+p[15]*t[2]))))+ebdy[i]-p[16])))+
(Y43[i]-w48)*log((((p[13]/(1+exp(-1*(p[14]+p[15]*t[3]))))+ebdy[i]-p[16])))+
(Y44[i]-w48)*log((((p[13]/(1+exp(-1*(p[14]+p[15]*t[4]))))+ebdy[i]-p[16])))+
(Y45[i]-w48)*log((((p[13]/(1+exp(-1*(p[14]+p[15]*t[5]))))+ebdy[i]-p[16])))+
(Y46[i]-w48)*log((((p[13]/(1+exp(-1*(p[14]+p[15]*t[6]))))+ebdy[i]-p[16])))+
(Y47[i]-w48)*log((((p[13]/(1+exp(-1*(p[14]+p[15]*t[7]))))+ebdy[i]-p[16])))-
lgamma(w48+1)-lgamma(Y41[i]-w48+1)-lgamma(Y42[i]-w48+1)-
lgamma(Y43[i]-w48+1)-lgamma(Y44[i]-w48+1)-lgamma(Y45[i]-w48+1)-
lgamma(Y46[i]-w48+1)-lgamma(Y47[i]-w48+1)
s48num=s48num+w48*exp(ls48)
s48den=s48den+exp(ls48)
}
s48[i]=s48num/s48den
}
s8=cbind(s18,s28,s38,s48)
s8
}

##### Função do -log(verossimilhança) considerando ew8dby
##### como verdadeira realização de w8
mlnv <- function(p) {
  lnvero=matrix(0,nrow=n,ncol=1)
  for (i in 1:n){
    lfxk= (-1*p[4])+log((p[4])**ew18[i])+
(-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[1]))))+xk*p[17]-p[4]))+
(-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[2]))))+xk*p[17]-p[4]))+
(-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[3]))))+xk*p[17]-p[4]))+
(-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[4]))))+xk*p[17]-p[4]))+
(-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[5]))))+xk*p[17]-p[4]))+
(-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[6]))))+xk*p[17]-p[4]))+
(-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[7]))))+xk*p[17]-p[4]))+
(Y11[i]-ew18[i])*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[1]))))+xk*p[17]-p[4])))+
(Y12[i]-ew18[i])*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[2]))))+xk*p[17]-p[4])))+
(Y13[i]-ew18[i])*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[3]))))+xk*p[17]-p[4])))+
(Y14[i]-ew18[i])*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[4]))))+xk*p[17]-p[4])))+
(Y15[i]-ew18[i])*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[5]))))+xk*p[17]-p[4])))+

```



```

(Y16[i]-ew18[i])*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[6]))))+xk*p[17]-p[4])))+
(Y17[i]-ew18[i])*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[7]))))+xk*p[17]-p[4])))-
lgamma(ew18[i]+1)-lgamma(Y11[i]-ew18[i]+1)-lgamma(Y12[i]-ew18[i]+1)-
lgamma(Y13[i]-ew18[i]+1)-lgamma(Y14[i]-ew18[i]+1)-lgamma(Y15[i]-ew18[i]+1)-
lgamma(Y16[i]-ew18[i]+1)-lgamma(Y17[i]-ew18[i]+1)+
(-1*p[8])+log((p[8])**ew28[i])+
(-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[1]))))+xk*p[17]-p[8]))+
(-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[2]))))+xk*p[17]-p[8]))+
(-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[3]))))+xk*p[17]-p[8]))+
(-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[4]))))+xk*p[17]-p[8]))+
(-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[5]))))+xk*p[17]-p[8]))+
(-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[6]))))+xk*p[17]-p[8]))+
(-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[7]))))+xk*p[17]-p[8]))+
(Y21[i]-ew28[i])*log((((p[5]/(1+exp(-1*(p[6]+p[7]*t[1]))))+xk*p[17]-p[8])))+
(Y22[i]-ew28[i])*log((((p[5]/(1+exp(-1*(p[6]+p[7]*t[2]))))+xk*p[17]-p[8])))+
(Y23[i]-ew28[i])*log((((p[5]/(1+exp(-1*(p[6]+p[7]*t[3]))))+xk*p[17]-p[8])))+
(Y24[i]-ew28[i])*log((((p[5]/(1+exp(-1*(p[6]+p[7]*t[4]))))+xk*p[17]-p[8])))+
(Y25[i]-ew28[i])*log((((p[5]/(1+exp(-1*(p[6]+p[7]*t[5]))))+xk*p[17]-p[8])))+
(Y26[i]-ew28[i])*log((((p[5]/(1+exp(-1*(p[6]+p[7]*t[6]))))+xk*p[17]-p[8])))+
(Y27[i]-ew28[i])*log((((p[5]/(1+exp(-1*(p[6]+p[7]*t[7]))))+xk*p[17]-p[8])))-
lgamma(ew28[i]+1)-lgamma(Y21[i]-ew28[i]+1)-lgamma(Y22[i]-ew28[i]+1)-
lgamma(Y23[i]-ew28[i]+1)-lgamma(Y24[i]-ew28[i]+1)-lgamma(Y25[i]-ew28[i]+1)-
lgamma(Y26[i]-ew28[i]+1)-lgamma(Y27[i]-ew28[i]+1)+
(-1*p[12])+log((p[12])**ew38[i])+
(-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[1]))))+xk*p[17]-p[12]))+
(-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[2]))))+xk*p[17]-p[12]))+
(-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[3]))))+xk*p[17]-p[12]))+
(-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[4]))))+xk*p[17]-p[12]))+
(-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[5]))))+xk*p[17]-p[12]))+
(-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[6]))))+xk*p[17]-p[12]))+
(-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[7]))))+xk*p[17]-p[12]))+
(Y31[i]-ew38[i])*log((((p[9]/(1+exp(-1*(p[10]+p[11]*t[1]))))+xk*p[17]-p[12])))+
(Y32[i]-ew38[i])*log((((p[9]/(1+exp(-1*(p[10]+p[11]*t[2]))))+xk*p[17]-p[12])))+
(Y33[i]-ew38[i])*log((((p[9]/(1+exp(-1*(p[10]+p[11]*t[3]))))+xk*p[17]-p[12])))+
(Y34[i]-ew38[i])*log((((p[9]/(1+exp(-1*(p[10]+p[11]*t[4]))))+xk*p[17]-p[12])))+
(Y35[i]-ew38[i])*log((((p[9]/(1+exp(-1*(p[10]+p[11]*t[5]))))+xk*p[17]-p[12])))+
(Y36[i]-ew38[i])*log((((p[9]/(1+exp(-1*(p[10]+p[11]*t[6]))))+xk*p[17]-p[12])))+
(Y37[i]-ew38[i])*log((((p[9]/(1+exp(-1*(p[10]+p[11]*t[7]))))+xk*p[17]-p[12])))-
lgamma(ew38[i]+1)-lgamma(Y31[i]-ew38[i]+1)-lgamma(Y32[i]-ew38[i]+1)-
lgamma(Y33[i]-ew38[i]+1)-lgamma(Y34[i]-ew38[i]+1)-lgamma(Y35[i]-ew38[i]+1)-
lgamma(Y36[i]-ew38[i]+1)-lgamma(Y37[i]-ew38[i]+1)+
(-1*p[16])+log((p[16])**ew48[i])+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[1]))))+xk*p[17]-p[16]))+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[2]))))+xk*p[17]-p[16]))+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[3]))))+xk*p[17]-p[16]))+

```

```

(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[4]))))+xk*p[17]-p[16]))+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[5]))))+xk*p[17]-p[16]))+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[6]))))+xk*p[17]-p[16]))+
(Y41[i]-ew48[i])*log((((p[13]/(1+exp(-1*(p[14]+p[15]*t[1]))))+xk*p[17]-p[16])))+
(Y42[i]-ew48[i])*log((((p[13]/(1+exp(-1*(p[14]+p[15]*t[2]))))+xk*p[17]-p[16])))+
(Y43[i]-ew48[i])*log((((p[13]/(1+exp(-1*(p[14]+p[15]*t[3]))))+xk*p[17]-p[16])))+
(Y44[i]-ew48[i])*log((((p[13]/(1+exp(-1*(p[14]+p[15]*t[4]))))+xk*p[17]-p[16])))+
(Y45[i]-ew48[i])*log((((p[13]/(1+exp(-1*(p[14]+p[15]*t[5]))))+xk*p[17]-p[16])))+
(Y46[i]-ew48[i])*log((((p[13]/(1+exp(-1*(p[14]+p[15]*t[6]))))+xk*p[17]-p[16])))+
(Y47[i]-ew48[i])*log((((p[13]/(1+exp(-1*(p[14]+p[15]*t[7]))))+xk*p[17]-p[16])))-
lgamma(ew48[i]+1)-lgamma(Y41[i]-ew48[i]+1)-lgamma(Y42[i]-ew48[i]+1)-
lgamma(Y43[i]-ew48[i]+1)-lgamma(Y44[i]-ew48[i]+1)-lgamma(Y45[i]-ew48[i]+1)-
lgamma(Y46[i]-ew48[i]+1)-lgamma(Y47[i]-ew48[i]+1)-lgamma(1/p[17])+
(1/p[17])*log(1/p[17])+
((1/p[17])-1)*log(xk*p[17])+log(p[17])
Akfxk=Ak*exp(lfxk)
lnvero[i]=log(sum(Akfxk))
}
lnveroM=sum(lnvero)
-lnveroM
}

```

```

##### Definição do gradiente da função a ser minimizada #####
##### 1)Vetor com -derivada((Ak*fxx)) para cada observação #####
dAkfxk <- deriv(
  ~-(Ak*exp((-1*p4)+log((p4)**ew18i))+
    (-1*((p1/(1+exp(-1*(p2+p3*t1))))+xk*p17-p4))+
    (-1*((p1/(1+exp(-1*(p2+p3*t2))))+xk*p17-p4))+
    (-1*((p1/(1+exp(-1*(p2+p3*t3))))+xk*p17-p4))+
    (-1*((p1/(1+exp(-1*(p2+p3*t4))))+xk*p17-p4))+
    (-1*((p1/(1+exp(-1*(p2+p3*t5))))+xk*p17-p4))+
    (-1*((p1/(1+exp(-1*(p2+p3*t6))))+xk*p17-p4))+
    (-1*((p1/(1+exp(-1*(p2+p3*t7))))+xk*p17-p4))+
    (Y11i-ew18i)*log((((p1/(1+exp(-1*(p2+p3*t1))))+xk*p17-p4))))+
    (Y12i-ew18i)*log((((p1/(1+exp(-1*(p2+p3*t2))))+xk*p17-p4))))+
    (Y13i-ew18i)*log((((p1/(1+exp(-1*(p2+p3*t3))))+xk*p17-p4))))+
    (Y14i-ew18i)*log((((p1/(1+exp(-1*(p2+p3*t4))))+xk*p17-p4))))+
    (Y15i-ew18i)*log((((p1/(1+exp(-1*(p2+p3*t5))))+xk*p17-p4))))+
    (Y16i-ew18i)*log((((p1/(1+exp(-1*(p2+p3*t6))))+xk*p17-p4))))+
    (Y17i-ew18i)*log((((p1/(1+exp(-1*(p2+p3*t7))))+xk*p17-p4))))-
    lgamma(ew18i+1)-lgamma(Y11i-ew18i+1)-lgamma(Y12i-ew18i+1)-
    lgamma(Y13i-ew18i+1)-lgamma(Y14i-ew18i+1)-lgamma(Y15i-ew18i+1)-
    lgamma(Y16i-ew18i+1)-lgamma(Y17i-ew18i+1)+
    (-1*p8)+log((p8)**ew28i)+
    (-1*((p5/(1+exp(-1*(p6+p7*t1))))+xk*p17-p8))+

```

$$\begin{aligned}
& (-1*((p5/(1+\exp(-1*(p6+p7*t2))))+xk*p17-p8))+ \\
& (-1*((p5/(1+\exp(-1*(p6+p7*t3))))+xk*p17-p8))+ \\
& (-1*((p5/(1+\exp(-1*(p6+p7*t4))))+xk*p17-p8))+ \\
& (-1*((p5/(1+\exp(-1*(p6+p7*t5))))+xk*p17-p8))+ \\
& (-1*((p5/(1+\exp(-1*(p6+p7*t6))))+xk*p17-p8))+ \\
& (-1*((p5/(1+\exp(-1*(p6+p7*t7))))+xk*p17-p8))+ \\
& (Y21i-ew28i)*\log(((p5/(1+\exp(-1*(p6+p7*t1))))+xk*p17-p8))+ \\
& (Y22i-ew28i)*\log(((p5/(1+\exp(-1*(p6+p7*t2))))+xk*p17-p8))+ \\
& (Y23i-ew28i)*\log(((p5/(1+\exp(-1*(p6+p7*t3))))+xk*p17-p8))+ \\
& (Y24i-ew28i)*\log(((p5/(1+\exp(-1*(p6+p7*t4))))+xk*p17-p8))+ \\
& (Y25i-ew28i)*\log(((p5/(1+\exp(-1*(p6+p7*t5))))+xk*p17-p8))+ \\
& (Y26i-ew28i)*\log(((p5/(1+\exp(-1*(p6+p7*t6))))+xk*p17-p8))+ \\
& (Y27i-ew28i)*\log(((p5/(1+\exp(-1*(p6+p7*t7))))+xk*p17-p8))- \\
& \lgamma(ew28i+1)-\lgamma(Y21i-ew28i+1)-\lgamma(Y22i-ew28i+1)- \\
& \lgamma(Y23i-ew28i+1)-\lgamma(Y24i-ew28i+1)-\lgamma(Y25i-ew28i+1)- \\
& \lgamma(Y26i-ew28i+1)-\lgamma(Y27i-ew28i+1)+ \\
& (-1*p12)+\log((p12)**ew38i)+ \\
& (-1*((p9/(1+\exp(-1*(p10+p11*t1))))+xk*p17-p12))+ \\
& (-1*((p9/(1+\exp(-1*(p10+p11*t2))))+xk*p17-p12))+ \\
& (-1*((p9/(1+\exp(-1*(p10+p11*t3))))+xk*p17-p12))+ \\
& (-1*((p9/(1+\exp(-1*(p10+p11*t4))))+xk*p17-p12))+ \\
& (-1*((p9/(1+\exp(-1*(p10+p11*t5))))+xk*p17-p12))+ \\
& (-1*((p9/(1+\exp(-1*(p10+p11*t6))))+xk*p17-p12))+ \\
& (-1*((p9/(1+\exp(-1*(p10+p11*t7))))+xk*p17-p12))+ \\
& (Y31i-ew38i)*\log(((p9/(1+\exp(-1*(p10+p11*t1))))+xk*p17-p12))+ \\
& (Y32i-ew38i)*\log(((p9/(1+\exp(-1*(p10+p11*t2))))+xk*p17-p12))+ \\
& (Y33i-ew38i)*\log(((p9/(1+\exp(-1*(p10+p11*t3))))+xk*p17-p12))+ \\
& (Y34i-ew38i)*\log(((p9/(1+\exp(-1*(p10+p11*t4))))+xk*p17-p12))+ \\
& (Y35i-ew38i)*\log(((p9/(1+\exp(-1*(p10+p11*t5))))+xk*p17-p12))+ \\
& (Y36i-ew38i)*\log(((p9/(1+\exp(-1*(p10+p11*t6))))+xk*p17-p12))+ \\
& (Y37i-ew38i)*\log(((p9/(1+\exp(-1*(p10+p11*t7))))+xk*p17-p12))- \\
& \lgamma(ew38i+1)-\lgamma(Y31i-ew38i+1)-\lgamma(Y32i-ew38i+1)- \\
& \lgamma(Y33i-ew38i+1)-\lgamma(Y34i-ew38i+1)-\lgamma(Y35i-ew38i+1)- \\
& \lgamma(Y36i-ew38i+1)-\lgamma(Y37i-ew38i+1)+ \\
& (-1*p16)+\log((p16)**ew48i)+ \\
& (-1*((p13/(1+\exp(-1*(p14+p15*t1))))+xk*p17-p16))+ \\
& (-1*((p13/(1+\exp(-1*(p14+p15*t2))))+xk*p17-p16))+ \\
& (-1*((p13/(1+\exp(-1*(p14+p15*t3))))+xk*p17-p16))+ \\
& (-1*((p13/(1+\exp(-1*(p14+p15*t4))))+xk*p17-p16))+ \\
& (-1*((p13/(1+\exp(-1*(p14+p15*t5))))+xk*p17-p16))+ \\
& (-1*((p13/(1+\exp(-1*(p14+p15*t6))))+xk*p17-p16))+ \\
& (-1*((p13/(1+\exp(-1*(p14+p15*t7))))+xk*p17-p16))+ \\
& (Y41i-ew48i)*\log(((p13/(1+\exp(-1*(p14+p15*t1))))+xk*p17-p16))+ \\
& (Y42i-ew48i)*\log(((p13/(1+\exp(-1*(p14+p15*t2))))+xk*p17-p16))+ \\
& (Y43i-ew48i)*\log(((p13/(1+\exp(-1*(p14+p15*t3))))+xk*p17-p16))+
\end{aligned}$$

```

(Y44i-ew48i)*log((((p13/(1+exp(-1*(p14+p15*t4))))+xk*p17-p16))))+
(Y45i-ew48i)*log((((p13/(1+exp(-1*(p14+p15*t5))))+xk*p17-p16))))+
(Y46i-ew48i)*log((((p13/(1+exp(-1*(p14+p15*t6))))+xk*p17-p16))))+
(Y47i-ew48i)*log((((p13/(1+exp(-1*(p14+p15*t7))))+xk*p17-p16))))-
  lgamma(ew48i+1)-lgamma(Y41i-ew48i+1)-lgamma(Y42i-ew48i+1)-
  lgamma(Y43i-ew48i+1)-lgamma(Y44i-ew48i+1)-lgamma(Y45i-ew48i+1)-
  lgamma(Y46i-ew48i+1)-lgamma(Y47i-ew48i+1)-lgamma(1/p17)+(1/p17)*log(1/p17)+
  ((1/p17)-1)*log(xk*p17+log(p17))),
c("p1","p2","p3","p4","p5","p6","p7","p8","p9","p10","p11","p12","p13",
  "p14","p15","p16","p17"),function(p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11,
  p12,p13,p14,p15,p16,p17,Ak,xk,t1,t2,t3,t4,t5,t6,t7,ew18i,ew28i,ew38i,
  ew48i,Y11i,Y12i,Y13i,Y14i,Y15i,Y16i,Y17i,Y21i,Y22i,Y23i,Y24i,Y25i,Y26i,
  Y27i,Y31i,Y32i,Y33i,Y34i,Y35i,Y36i,Y37i,Y41i,Y42i,Y43i,Y44i,Y45i,Y46i,
  Y47i) NULL)

##### 2)Vetor com gradiente para maximização de log(verossimilhança)
##### para cada observação
grd <- function(p) {
  p1=p[1]; p2=p[2]; p3=p[3]; p4=p[4]; p5=p[5]; p6=p[6]; p7=p[7]
  p8=p[8]; p9=p[9]; p10=p[10]; p11=p[11]; p12=p[12]; p13=p[13]
  p14=p[14]; p15=p[15]; p16=p[16]; p17=p[17]
  t1=t[1]; t2=t[2]; t3=t[3]; t4=t[4]; t5=t[5]; t6=t[6]; t7=t[7]
  dlnsAkfxki=matrix(0,nrow=length(Y11),ncol=length(p))
  numer=matrix(0,nrow=length(Y11),ncol=length(p))
  denom=matrix(0,nrow=length(Y11),ncol=1)
  for (i in 1:length(Y11)) {
    ew18i=ew18[i]; ew28i=ew28[i]; ew38i=ew38[i]; ew48i=ew48[i]
    Y11i=Y11[i]; Y12i=Y12[i]; Y13i=Y13[i]; Y14i=Y14[i]
    Y15i=Y15[i]; Y16i=Y16[i]; Y17i=Y17[i]; Y21i=Y21[i]
    Y22i=Y22[i]; Y23i=Y23[i]; Y24i=Y24[i]; Y25i=Y25[i]
    Y26i=Y26[i]; Y27i=Y27[i]; Y31i=Y31[i]; Y32i=Y32[i]
    Y33i=Y33[i]; Y34i=Y34[i]; Y35i=Y35[i]; Y36i=Y36[i]
    Y37i=Y37[i]; Y41i=Y41[i]; Y42i=Y42[i]; Y43i=Y43[i]
    Y44i=Y44[i]; Y45i=Y45[i]; Y46i=Y46[i]; Y47i=Y47[i]
    numer[i,]=colSums(attr(dAkfxk(p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11,p12,
      p13,p14,p15,p16,p17,Ak,xk,t1,t2,t3,t4,t5,t6,t7,ew18i,
      ew28i,ew38i,ew48i,Y11i,Y12i,Y13i,Y14i,Y15i,Y16i,
      Y17i,Y21i,Y22i,Y23i,Y24i,Y25i,Y26i,Y27i,Y31i,Y32i,
      Y33i,Y34i,Y35i,Y36i,Y37i,Y41i,Y42i,Y43i,Y44i,Y45i,
      Y46i,Y47i),"gradient"))
    denom[i]=sum(Ak*exp((-1*p4)+log((p4)**ew18i)+
      (-1*((p1/(1+exp(-1*(p2+p3*t1))))+xk*p17-p4))+
      (-1*((p1/(1+exp(-1*(p2+p3*t2))))+xk*p17-p4))+
      (-1*((p1/(1+exp(-1*(p2+p3*t3))))+xk*p17-p4))+
      (-1*((p1/(1+exp(-1*(p2+p3*t4))))+xk*p17-p4))+

```

$$\begin{aligned}
& (-1*((p1/(1+\exp(-1*(p2+p3*t5))))+xk*p17-p4))+ \\
& (-1*((p1/(1+\exp(-1*(p2+p3*t6))))+xk*p17-p4))+ \\
& (-1*((p1/(1+\exp(-1*(p2+p3*t7))))+xk*p17-p4))+ \\
& (Y11i-ew18i)*\log(((p1/(1+\exp(-1*(p2+p3*t1))))+xk*p17-p4)))+ \\
& (Y12i-ew18i)*\log(((p1/(1+\exp(-1*(p2+p3*t2))))+xk*p17-p4)))+ \\
& (Y13i-ew18i)*\log(((p1/(1+\exp(-1*(p2+p3*t3))))+xk*p17-p4)))+ \\
& (Y14i-ew18i)*\log(((p1/(1+\exp(-1*(p2+p3*t4))))+xk*p17-p4)))+ \\
& (Y15i-ew18i)*\log(((p1/(1+\exp(-1*(p2+p3*t5))))+xk*p17-p4)))+ \\
& (Y16i-ew18i)*\log(((p1/(1+\exp(-1*(p2+p3*t6))))+xk*p17-p4)))+ \\
& (Y17i-ew18i)*\log(((p1/(1+\exp(-1*(p2+p3*t7))))+xk*p17-p4))- \\
& \lgamma(ew18i+1)-\lgamma(Y11i-ew18i+1)-\lgamma(Y12i-ew18i+1)- \\
& \lgamma(Y13i-ew18i+1)-\lgamma(Y14i-ew18i+1)- \\
& \lgamma(Y15i-ew18i+1)-\lgamma(Y16i-ew18i+1)- \\
& \lgamma(Y17i-ew18i+1)+ \\
& (-1*p8)+\log((p8)**ew28i)+ \\
& (-1*((p5/(1+\exp(-1*(p6+p7*t1))))+xk*p17-p8))+ \\
& (-1*((p5/(1+\exp(-1*(p6+p7*t2))))+xk*p17-p8))+ \\
& (-1*((p5/(1+\exp(-1*(p6+p7*t3))))+xk*p17-p8))+ \\
& (-1*((p5/(1+\exp(-1*(p6+p7*t4))))+xk*p17-p8))+ \\
& (-1*((p5/(1+\exp(-1*(p6+p7*t5))))+xk*p17-p8))+ \\
& (-1*((p5/(1+\exp(-1*(p6+p7*t6))))+xk*p17-p8))+ \\
& (-1*((p5/(1+\exp(-1*(p6+p7*t7))))+xk*p17-p8))+ \\
& (Y21i-ew28i)*\log(((p5/(1+\exp(-1*(p6+p7*t1))))+xk*p17-p8)))+ \\
& (Y22i-ew28i)*\log(((p5/(1+\exp(-1*(p6+p7*t2))))+xk*p17-p8)))+ \\
& (Y23i-ew28i)*\log(((p5/(1+\exp(-1*(p6+p7*t3))))+xk*p17-p8)))+ \\
& (Y24i-ew28i)*\log(((p5/(1+\exp(-1*(p6+p7*t4))))+xk*p17-p8)))+ \\
& (Y25i-ew28i)*\log(((p5/(1+\exp(-1*(p6+p7*t5))))+xk*p17-p8)))+ \\
& (Y26i-ew28i)*\log(((p5/(1+\exp(-1*(p6+p7*t6))))+xk*p17-p8)))+ \\
& (Y27i-ew28i)*\log(((p5/(1+\exp(-1*(p6+p7*t7))))+xk*p17-p8))- \\
& \lgamma(ew28i+1)-\lgamma(Y21i-ew28i+1)-\lgamma(Y22i-ew28i+1)- \\
& \lgamma(Y23i-ew28i+1)-\lgamma(Y24i-ew28i+1)- \\
& \lgamma(Y25i-ew28i+1)-\lgamma(Y26i-ew28i+1)- \\
& \lgamma(Y27i-ew28i+1)+ \\
& (-1*p12)+\log((p12)**ew38i)+ \\
& (-1*((p9/(1+\exp(-1*(p10+p11*t1))))+xk*p17-p12))+ \\
& (-1*((p9/(1+\exp(-1*(p10+p11*t2))))+xk*p17-p12))+ \\
& (-1*((p9/(1+\exp(-1*(p10+p11*t3))))+xk*p17-p12))+ \\
& (-1*((p9/(1+\exp(-1*(p10+p11*t4))))+xk*p17-p12))+ \\
& (-1*((p9/(1+\exp(-1*(p10+p11*t5))))+xk*p17-p12))+ \\
& (-1*((p9/(1+\exp(-1*(p10+p11*t6))))+xk*p17-p12))+ \\
& (-1*((p9/(1+\exp(-1*(p10+p11*t7))))+xk*p17-p12))+ \\
& (Y31i-ew38i)*\log(((p9/(1+\exp(-1*(p10+p11*t1))))+xk*p17-p12)))+ \\
& (Y32i-ew38i)*\log(((p9/(1+\exp(-1*(p10+p11*t2))))+xk*p17-p12)))+ \\
& (Y33i-ew38i)*\log(((p9/(1+\exp(-1*(p10+p11*t3))))+xk*p17-p12)))+ \\
& (Y34i-ew38i)*\log(((p9/(1+\exp(-1*(p10+p11*t4))))+xk*p17-p12)))+
\end{aligned}$$

```

(Y35i-ew38i)*log(((p9/(1+exp(-1*(p10+p11*t5))))+xk*p17-p12))+
(Y36i-ew38i)*log(((p9/(1+exp(-1*(p10+p11*t6))))+xk*p17-p12))+
(Y37i-ew38i)*log(((p9/(1+exp(-1*(p10+p11*t7))))+xk*p17-p12))-
lgamma(ew38i+1)-lgamma(Y31i-ew38i+1)-lgamma(Y32i-ew38i+1)-
lgamma(Y33i-ew38i+1)-lgamma(Y34i-ew38i+1)-
lgamma(Y35i-ew38i+1)-lgamma(Y36i-ew38i+1)-
lgamma(Y37i-ew38i+1)+
(-1*p16)+log((p16)**ew48i)+
(-1*((p13/(1+exp(-1*(p14+p15*t1))))+xk*p17-p16))+
(-1*((p13/(1+exp(-1*(p14+p15*t2))))+xk*p17-p16))+
(-1*((p13/(1+exp(-1*(p14+p15*t3))))+xk*p17-p16))+
(-1*((p13/(1+exp(-1*(p14+p15*t4))))+xk*p17-p16))+
(-1*((p13/(1+exp(-1*(p14+p15*t5))))+xk*p17-p16))+
(-1*((p13/(1+exp(-1*(p14+p15*t6))))+xk*p17-p16))+
(-1*((p13/(1+exp(-1*(p14+p15*t7))))+xk*p17-p16))+
(Y41i-ew48i)*log(((p13/(1+exp(-1*(p14+p15*t1))))+xk*p17-p16))+
(Y42i-ew48i)*log(((p13/(1+exp(-1*(p14+p15*t2))))+xk*p17-p16))+
(Y43i-ew48i)*log(((p13/(1+exp(-1*(p14+p15*t3))))+xk*p17-p16))+
(Y44i-ew48i)*log(((p13/(1+exp(-1*(p14+p15*t4))))+xk*p17-p16))+
(Y45i-ew48i)*log(((p13/(1+exp(-1*(p14+p15*t5))))+xk*p17-p16))+
(Y46i-ew48i)*log(((p13/(1+exp(-1*(p14+p15*t6))))+xk*p17-p16))+
(Y47i-ew48i)*log(((p13/(1+exp(-1*(p14+p15*t7))))+xk*p17-p16))-
lgamma(ew48i+1)-lgamma(Y41i-ew48i+1)-lgamma(Y42i-ew48i+1)-
lgamma(Y43i-ew48i+1)-lgamma(Y44i-ew48i+1)-
lgamma(Y45i-ew48i+1)-lgamma(Y46i-ew48i+1)-
lgamma(Y47i-ew48i+1)-lgamma(1/p17)+
(1/p17)*log(1/p17)+((1/p17)-1)*log(xk*p17)+log(p17)))
dlnsAkfxki[i,]=numer[i,]/denom[i]
}
dlnsAkfxk=colSums(dlnsAkfxki)
dlnsAkfxk
}

```

Início do Processo de Simulação

```

W11=matrix(0,nrow=n,ncol=1); W12=matrix(0,nrow=n,ncol=1)
W13=matrix(0,nrow=n,ncol=1); W14=matrix(0,nrow=n,ncol=1)
W15=matrix(0,nrow=n,ncol=1); W16=matrix(0,nrow=n,ncol=1)
W17=matrix(0,nrow=n,ncol=1); W18=matrix(0,nrow=n,ncol=1)

W21=matrix(0,nrow=n,ncol=1); W22=matrix(0,nrow=n,ncol=1)
W23=matrix(0,nrow=n,ncol=1); W24=matrix(0,nrow=n,ncol=1)
W25=matrix(0,nrow=n,ncol=1); W26=matrix(0,nrow=n,ncol=1)
W27=matrix(0,nrow=n,ncol=1); W28=matrix(0,nrow=n,ncol=1)

```



```

W31=matrix(0,nrow=n,ncol=1); W32=matrix(0,nrow=n,ncol=1)
W33=matrix(0,nrow=n,ncol=1); W34=matrix(0,nrow=n,ncol=1)
W35=matrix(0,nrow=n,ncol=1); W36=matrix(0,nrow=n,ncol=1)
W37=matrix(0,nrow=n,ncol=1); W38=matrix(0,nrow=n,ncol=1)

W41=matrix(0,nrow=n,ncol=1); W42=matrix(0,nrow=n,ncol=1)
W43=matrix(0,nrow=n,ncol=1); W44=matrix(0,nrow=n,ncol=1)
W45=matrix(0,nrow=n,ncol=1); W46=matrix(0,nrow=n,ncol=1)
W47=matrix(0,nrow=n,ncol=1); W48=matrix(0,nrow=n,ncol=1)

set.seed(33); smt=runif(1000)
smt=round(smt,digits=5); smt=smt*100000
result=matrix(0,nrow=2000,ncol=24)
nsimul=1; k=1
system.time(
while (nsimul <=1000){
set.seed(smt[nsimul])

##### Gerando os efeitos aleatórios (bi)
##### seguindo distribuição Gama(1/sigma2b,1/sigma2b)
##### Obs: shape = primeiro parâmetro da gama,
##### scale = 1/(segundo parâmetro da gama)

# Efeito aleatório para o bloco 1
b1=rgamma(1, shape=1/sigma2bp, scale=sigma2bp)
# Efeito aleatório para o bloco 2
b2=rgamma(1, shape=1/sigma2bp, scale=sigma2bp)
# Efeito aleatório para o bloco 3
b3=rgamma(1, shape=1/sigma2bp, scale=sigma2bp)
# Efeito aleatório para o bloco 4
b4=rgamma(1, shape=1/sigma2bp, scale=sigma2bp)
# Efeito aleatório para o bloco 5
b5=rgamma(1, shape=1/sigma2bp, scale=sigma2bp)

b=c(b1,b2,b3,b4,b5)

##### Gerando os valores das variáveis latentes
##### condicionais aos valores de bi
for (i in 1:n) {

#Variáveis latentes para o tratamento 1
W11[i]=rpois(1,mu1[1]+b[i]-theta18p); W12[i]=rpois(1,mu1[2]+b[i]-theta18p)
W13[i]=rpois(1,mu1[3]+b[i]-theta18p); W14[i]=rpois(1,mu1[4]+b[i]-theta18p)
W15[i]=rpois(1,mu1[5]+b[i]-theta18p); W16[i]=rpois(1,mu1[6]+b[i]-theta18p)
W17[i]=rpois(1,mu1[7]+b[i]-theta18p); W18[i]=rpois(1,theta18p)

```

```

#Variáveis latentes para o tratamento 2
W21[i]=rpois(1,mu2[1]+b[i]-theta28p); W22[i]=rpois(1,mu2[2]+b[i]-theta28p)
W23[i]=rpois(1,mu2[3]+b[i]-theta28p); W24[i]=rpois(1,mu2[4]+b[i]-theta28p)
W25[i]=rpois(1,mu2[5]+b[i]-theta28p); W26[i]=rpois(1,mu2[6]+b[i]-theta28p)
W27[i]=rpois(1,mu2[7]+b[i]-theta28p); W28[i]=rpois(1,theta28p)

#Variáveis latentes para o tratamento 3
W31[i]=rpois(1,mu3[1]+b[i]-theta38p); W32[i]=rpois(1,mu3[2]+b[i]-theta38p)
W33[i]=rpois(1,mu3[3]+b[i]-theta38p); W34[i]=rpois(1,mu3[4]+b[i]-theta38p)
W35[i]=rpois(1,mu3[5]+b[i]-theta38p); W36[i]=rpois(1,mu3[6]+b[i]-theta38p)
W37[i]=rpois(1,mu3[7]+b[i]-theta38p); W38[i]=rpois(1,theta38p)

#Variáveis latentes para o tratamento 4
W41[i]=rpois(1,mu4[1]+b[i]-theta48p); W42[i]=rpois(1,mu4[2]+b[i]-theta48p)
W43[i]=rpois(1,mu4[3]+b[i]-theta48p); W44[i]=rpois(1,mu4[4]+b[i]-theta48p)
W45[i]=rpois(1,mu4[5]+b[i]-theta48p); W46[i]=rpois(1,mu4[6]+b[i]-theta48p)
W47[i]=rpois(1,mu4[7]+b[i]-theta48p); W48[i]=rpois(1,theta48p)
}

#### Gerando os valores observados condicionais aos valores de Zij #####
#Variáveis observadas para o tratamento 1
Y11=W11+W18; Y12=W12+W18; Y13=W13+W18; Y14=W14+W18
Y15=W15+W18; Y16=W16+W18; Y17=W17+W18;

Y1=cbind(Y11,Y12,Y13,Y14,Y15,Y16,Y17)
Ytrat1=rbind(t(t(Y11)),t(t(Y12)),t(t(Y13)),t(t(Y14)),
             t(t(Y15)),t(t(Y16)),t(t(Y17)))

#Variáveis observadas para o tratamento 2
Y21=W21+W28; Y22=W22+W28; Y23=W23+W28; Y24=W24+W28
Y25=W25+W28; Y26=W26+W28; Y27=W27+W28

Y2=cbind(Y21,Y22,Y23,Y24,Y25,Y26,Y27)
Ytrat2=rbind(t(t(Y21)),t(t(Y22)),t(t(Y23)),t(t(Y24)),
             t(t(Y25)),t(t(Y26)),t(t(Y27)))

#Variáveis observadas para o tratamento 3
Y31=W31+W38; Y32=W32+W38; Y33=W33+W38; Y34=W34+W38
Y35=W35+W38; Y36=W36+W38; Y37=W37+W38

Y3=cbind(Y31,Y32,Y33,Y34,Y35,Y36,Y37)
Ytrat3=rbind(t(t(Y31)),t(t(Y32)),t(t(Y33)),t(t(Y34)),
             t(t(Y35)),t(t(Y36)),t(t(Y37)))

#Variáveis observadas para o tratamento 4

```



```

Y41=W41+W48; Y42=W42+W48; Y43=W43+W48; Y44=W44+W48
Y45=W45+W48; Y46=W46+W48; Y47=W47+W48

Y4=cbind(Y41,Y42,Y43,Y44,Y45,Y46,Y47)
Ytrat4=rbind(t(t(Y41)),t(t(Y42)),t(t(Y43)),t(t(Y44)),
              t(t(Y45)),t(t(Y46)),t(t(Y47)))

Y=rbind(Y1,Y2,Y3,Y4)
Yt1=Y[,1]; Yt2=Y[,2]; Yt3=Y[,3]; Yt4=Y[,4]
Yt5=Y[,5]; Yt6=Y[,6]; Yt7=Y[,7]

tempo=c(rep(t[1],n),rep(t[2],n),rep(t[3],n),rep(t[4],n),rep(t[5],n),
         rep(t[6],n),rep(t[7],n))
Ytrat=data.frame(tempo,Ytrat1,Ytrat2,Ytrat3,Ytrat4)

#Encontrando valores iniciais para os parâmetros da regressão
init1 <- nlreg( Ytrat1 ~ (alfa1/(1+exp(-1*(delta1+gama1*tempo))))),
              start = c(alfa1=200, delta1=-1, gama1=0.05),
              data = Ytrat)
init2 <- nlreg( Ytrat2 ~ (alfa2/(1+exp(-1*(delta2+gama2*tempo))))),
              start = c(alfa2=200, delta2=-1, gama2=0.05),
              data = Ytrat)
init3 <- nlreg( Ytrat3 ~ (alfa3/(1+exp(-1*(delta3+gama3*tempo))))),
              start = c(alfa3=200, delta3=-1, gama3=0.05),
              data = Ytrat)
init4 <- nlreg( Ytrat4 ~ (alfa4/(1+exp(-1*(delta4+gama4*tempo))))),
              start = c(alfa4=200, delta4=-1, gama4=0.05),
              data = Ytrat)

alfa1=init1$coef[1]; delta1=init1$coef[2]; gama1=init1$coef[3]
alfa1_ini=alfa1; delta1_ini=delta1; gama1_ini=gama1
alfa2=init2$coef[1]; delta2=init2$coef[2]; gama2=init2$coef[3]
alfa2_ini=alfa2; delta2_ini=delta2; gama2_ini=gama2
alfa3=init3$coef[1]; delta3=init3$coef[2]; gama3=init3$coef[3]
alfa3_ini=alfa3; delta3_ini=delta3; gama3_ini=gama3
alfa4=init4$coef[1]; delta4=init4$coef[2]; gama4=init4$coef[3]
alfa4_ini=alfa4; delta4_ini=delta4; gama4_ini=gama4

sigma2b=mean(var(c(Y[1,1],Y[6,1],Y[11,1],Y[16,1])),
              var(c(Y[1,2],Y[6,2],Y[11,2],Y[16,2])),
              var(c(Y[1,3],Y[6,3],Y[11,3],Y[16,3])),
              var(c(Y[1,4],Y[6,4],Y[11,4],Y[16,4])),
              var(c(Y[1,5],Y[6,5],Y[11,5],Y[16,5])),
              var(c(Y[1,6],Y[6,6],Y[11,6],Y[16,6])),
              var(c(Y[1,7],Y[6,7],Y[11,7],Y[16,7]))),

```

```

var(c(Y[2,1],Y[7,1],Y[12,1],Y[17,1])),
var(c(Y[2,2],Y[7,2],Y[12,2],Y[17,2])),
var(c(Y[2,3],Y[7,3],Y[12,3],Y[17,3])),
var(c(Y[2,4],Y[7,4],Y[12,4],Y[17,4])),
var(c(Y[2,5],Y[7,5],Y[12,5],Y[17,5])),
var(c(Y[2,6],Y[7,6],Y[12,6],Y[17,6])),
var(c(Y[2,7],Y[7,7],Y[12,7],Y[17,7])),
var(c(Y[3,1],Y[8,1],Y[13,1],Y[18,1])),
var(c(Y[3,2],Y[8,2],Y[13,2],Y[18,2])),
var(c(Y[3,3],Y[8,3],Y[13,3],Y[18,3])),
var(c(Y[3,4],Y[8,4],Y[13,4],Y[18,4])),
var(c(Y[3,5],Y[8,5],Y[13,5],Y[18,5])),
var(c(Y[3,6],Y[8,6],Y[13,6],Y[18,6])),
var(c(Y[3,7],Y[8,7],Y[13,7],Y[18,7])),
var(c(Y[4,1],Y[9,1],Y[14,1],Y[19,1])),
var(c(Y[4,2],Y[9,2],Y[14,2],Y[19,2])),
var(c(Y[4,3],Y[9,3],Y[14,3],Y[19,3])),
var(c(Y[4,4],Y[9,4],Y[14,4],Y[19,4])),
var(c(Y[4,5],Y[9,5],Y[14,5],Y[19,5])),
var(c(Y[4,6],Y[9,6],Y[14,6],Y[19,6])),
var(c(Y[4,7],Y[9,7],Y[14,7],Y[19,7])),
var(c(Y[5,1],Y[10,1],Y[15,1],Y[20,1])),
var(c(Y[5,2],Y[10,2],Y[15,2],Y[20,2])),
var(c(Y[5,3],Y[10,3],Y[15,3],Y[20,3])),
var(c(Y[5,4],Y[10,4],Y[15,4],Y[20,4])),
var(c(Y[5,5],Y[10,5],Y[15,5],Y[20,5])),
var(c(Y[5,6],Y[10,6],Y[15,6],Y[20,6])),
var(c(Y[5,7],Y[10,7],Y[15,7],Y[20,7]))
sigma2b_ini=sigma2b

#Encontrando os valores iniciais para os parâmetros de covariância
if (cov(Y11,Y12)>0) cov12_1=cov(Y11,Y12) else cov12_1=0
if (cov(Y11,Y13)>0) cov13_1=cov(Y11,Y13) else cov13_1=0
if (cov(Y11,Y14)>0) cov14_1=cov(Y11,Y14) else cov14_1=0
if (cov(Y11,Y15)>0) cov15_1=cov(Y11,Y15) else cov15_1=0
if (cov(Y11,Y16)>0) cov16_1=cov(Y11,Y16) else cov16_1=0
if (cov(Y11,Y17)>0) cov17_1=cov(Y11,Y17) else cov17_1=0
if (cov(Y12,Y13)>0) cov23_1=cov(Y12,Y13) else cov23_1=0
if (cov(Y12,Y14)>0) cov24_1=cov(Y12,Y14) else cov24_1=0
if (cov(Y12,Y15)>0) cov25_1=cov(Y12,Y15) else cov25_1=0
if (cov(Y12,Y16)>0) cov26_1=cov(Y12,Y16) else cov26_1=0
if (cov(Y12,Y17)>0) cov27_1=cov(Y12,Y17) else cov27_1=0
if (cov(Y13,Y14)>0) cov34_1=cov(Y13,Y14) else cov34_1=0
if (cov(Y13,Y15)>0) cov35_1=cov(Y13,Y15) else cov35_1=0
if (cov(Y13,Y16)>0) cov36_1=cov(Y13,Y16) else cov36_1=0

```

```

if (cov(Y13,Y17)>0) cov37_1=cov(Y13,Y17) else cov37_1=0
if (cov(Y14,Y15)>0) cov45_1=cov(Y14,Y15) else cov45_1=0
if (cov(Y14,Y16)>0) cov46_1=cov(Y14,Y16) else cov46_1=0
if (cov(Y14,Y17)>0) cov47_1=cov(Y14,Y17) else cov47_1=0
if (cov(Y15,Y16)>0) cov56_1=cov(Y15,Y16) else cov56_1=0
if (cov(Y15,Y17)>0) cov57_1=cov(Y15,Y17) else cov57_1=0
if (cov(Y16,Y17)>0) cov67_1=cov(Y16,Y17) else cov67_1=0

cov1=c(cov12_1,cov13_1,cov14_1,cov15_1,cov16_1,cov17_1,cov23_1,cov24_1,
      cov25_1,cov26_1,cov27_1,cov34_1,cov35_1,cov36_1,cov37_1,cov45_1,
      cov46_1,cov47_1,cov56_1,cov57_1,cov67_1)

theta18=theta18=min(mean(cov1),(alfa1/(1+exp(-1*(delta1+gama1*t[1])))),
                    (alfa1/(1+exp(-1*(delta1+gama1*t[2])))),
                    (alfa1/(1+exp(-1*(delta1+gama1*t[3])))),
                    (alfa1/(1+exp(-1*(delta1+gama1*t[4])))),
                    (alfa1/(1+exp(-1*(delta1+gama1*t[5])))),
                    (alfa1/(1+exp(-1*(delta1+gama1*t[6])))),
                    (alfa1/(1+exp(-1*(delta1+gama1*t[7])))))
theta18_ini=theta18

if (cov(Y21,Y22)>0) cov12_2=cov(Y21,Y22) else cov12_2=0
if (cov(Y21,Y23)>0) cov13_2=cov(Y21,Y23) else cov13_2=0
if (cov(Y21,Y24)>0) cov14_2=cov(Y21,Y24) else cov14_2=0
if (cov(Y21,Y25)>0) cov15_2=cov(Y21,Y25) else cov15_2=0
if (cov(Y21,Y26)>0) cov16_2=cov(Y21,Y26) else cov16_2=0
if (cov(Y21,Y27)>0) cov17_2=cov(Y21,Y27) else cov17_2=0
if (cov(Y22,Y23)>0) cov23_2=cov(Y22,Y23) else cov23_2=0
if (cov(Y22,Y24)>0) cov24_2=cov(Y22,Y24) else cov24_2=0
if (cov(Y22,Y25)>0) cov25_2=cov(Y22,Y25) else cov25_2=0
if (cov(Y22,Y26)>0) cov26_2=cov(Y22,Y26) else cov26_2=0
if (cov(Y22,Y27)>0) cov27_2=cov(Y22,Y27) else cov27_2=0
if (cov(Y23,Y24)>0) cov34_2=cov(Y23,Y24) else cov34_2=0
if (cov(Y23,Y25)>0) cov35_2=cov(Y23,Y25) else cov35_2=0
if (cov(Y23,Y26)>0) cov36_2=cov(Y23,Y26) else cov36_2=0
if (cov(Y23,Y27)>0) cov37_2=cov(Y23,Y27) else cov37_2=0
if (cov(Y24,Y25)>0) cov45_2=cov(Y24,Y25) else cov45_2=0
if (cov(Y24,Y26)>0) cov46_2=cov(Y24,Y26) else cov46_2=0
if (cov(Y24,Y27)>0) cov47_2=cov(Y24,Y27) else cov47_2=0
if (cov(Y25,Y26)>0) cov56_2=cov(Y25,Y26) else cov56_2=0
if (cov(Y25,Y27)>0) cov57_2=cov(Y25,Y27) else cov57_2=0
if (cov(Y26,Y27)>0) cov67_2=cov(Y26,Y27) else cov67_2=0

cov2=c(cov12_2,cov13_2,cov14_2,cov15_2,cov16_2,cov17_2,cov23_2,cov24_2,
      cov25_2,cov26_2,cov27_2,cov34_2,cov35_2,cov36_2,cov37_2,cov45_2,

```

```

cov46_2,cov47_2,cov56_2,cov57_2,cov67_2)

theta28=min(mean(cov2),(alfa2/(1+exp(-1*(delta2+gama2*t[1])))),
            (alfa2/(1+exp(-1*(delta2+gama2*t[2])))),
            (alfa2/(1+exp(-1*(delta2+gama2*t[3])))),
            (alfa2/(1+exp(-1*(delta2+gama2*t[4])))),
            (alfa2/(1+exp(-1*(delta2+gama2*t[5])))),
            (alfa2/(1+exp(-1*(delta2+gama2*t[6])))),
            (alfa2/(1+exp(-1*(delta2+gama2*t[7])))))
theta28_ini=theta28

if (cov(Y31,Y32)>0) cov12_3=cov(Y31,Y32) else cov12_3=0
if (cov(Y31,Y33)>0) cov13_3=cov(Y31,Y33) else cov13_3=0
if (cov(Y31,Y34)>0) cov14_3=cov(Y31,Y34) else cov14_3=0
if (cov(Y31,Y35)>0) cov15_3=cov(Y31,Y35) else cov15_3=0
if (cov(Y31,Y36)>0) cov16_3=cov(Y31,Y36) else cov16_3=0
if (cov(Y31,Y37)>0) cov17_3=cov(Y31,Y37) else cov17_3=0
if (cov(Y32,Y33)>0) cov23_3=cov(Y32,Y33) else cov23_3=0
if (cov(Y32,Y34)>0) cov24_3=cov(Y32,Y34) else cov24_3=0
if (cov(Y32,Y35)>0) cov25_3=cov(Y32,Y35) else cov25_3=0
if (cov(Y32,Y36)>0) cov26_3=cov(Y32,Y36) else cov26_3=0
if (cov(Y32,Y37)>0) cov27_3=cov(Y32,Y37) else cov27_3=0
if (cov(Y33,Y34)>0) cov34_3=cov(Y33,Y34) else cov34_3=0
if (cov(Y33,Y35)>0) cov35_3=cov(Y33,Y35) else cov35_3=0
if (cov(Y33,Y36)>0) cov36_3=cov(Y33,Y36) else cov36_3=0
if (cov(Y33,Y37)>0) cov37_3=cov(Y33,Y37) else cov37_3=0
if (cov(Y34,Y35)>0) cov45_3=cov(Y34,Y35) else cov45_3=0
if (cov(Y34,Y36)>0) cov46_3=cov(Y34,Y36) else cov46_3=0
if (cov(Y34,Y37)>0) cov47_3=cov(Y34,Y37) else cov47_3=0
if (cov(Y35,Y36)>0) cov56_3=cov(Y35,Y36) else cov56_3=0
if (cov(Y35,Y37)>0) cov57_3=cov(Y35,Y37) else cov57_3=0
if (cov(Y36,Y37)>0) cov67_3=cov(Y36,Y37) else cov67_3=0

cov3=c(cov12_3,cov13_3,cov14_3,cov15_3,cov16_3,cov17_3,cov23_3,cov24_3,
      cov25_3,cov26_3,cov27_3,cov34_3,cov35_3,cov36_3,cov37_3,cov45_3,
      cov46_3,cov47_3,cov56_3,cov57_3,cov67_3)

theta38=min(mean(cov3),(alfa3/(1+exp(-1*(delta3+gama3*t[1])))),
            (alfa3/(1+exp(-1*(delta3+gama3*t[2])))),
            (alfa3/(1+exp(-1*(delta3+gama3*t[3])))),
            (alfa3/(1+exp(-1*(delta3+gama3*t[4])))),
            (alfa3/(1+exp(-1*(delta3+gama3*t[5])))),
            (alfa3/(1+exp(-1*(delta3+gama3*t[6])))),
            (alfa3/(1+exp(-1*(delta3+gama3*t[7])))))
theta38_ini=theta38

```

```

if (cov(Y41,Y42)>0) cov12_4=cov(Y41,Y42) else cov12_4=0
if (cov(Y41,Y43)>0) cov13_4=cov(Y41,Y43) else cov13_4=0
if (cov(Y41,Y44)>0) cov14_4=cov(Y41,Y44) else cov14_4=0
if (cov(Y41,Y45)>0) cov15_4=cov(Y41,Y45) else cov15_4=0
if (cov(Y41,Y46)>0) cov16_4=cov(Y41,Y46) else cov16_4=0
if (cov(Y41,Y47)>0) cov17_4=cov(Y41,Y47) else cov17_4=0
if (cov(Y42,Y43)>0) cov23_4=cov(Y42,Y43) else cov23_4=0
if (cov(Y42,Y44)>0) cov24_4=cov(Y42,Y44) else cov24_4=0
if (cov(Y42,Y45)>0) cov25_4=cov(Y42,Y45) else cov25_4=0
if (cov(Y42,Y46)>0) cov26_4=cov(Y42,Y46) else cov26_4=0
if (cov(Y42,Y47)>0) cov27_4=cov(Y42,Y47) else cov27_4=0
if (cov(Y43,Y44)>0) cov34_4=cov(Y43,Y44) else cov34_4=0
if (cov(Y43,Y45)>0) cov35_4=cov(Y43,Y45) else cov35_4=0
if (cov(Y43,Y46)>0) cov36_4=cov(Y43,Y46) else cov36_4=0
if (cov(Y43,Y47)>0) cov37_4=cov(Y43,Y47) else cov37_4=0
if (cov(Y44,Y45)>0) cov45_4=cov(Y44,Y45) else cov45_4=0
if (cov(Y44,Y46)>0) cov46_4=cov(Y44,Y46) else cov46_4=0
if (cov(Y44,Y47)>0) cov47_4=cov(Y44,Y47) else cov47_4=0
if (cov(Y45,Y46)>0) cov56_4=cov(Y45,Y46) else cov56_4=0
if (cov(Y45,Y47)>0) cov57_4=cov(Y45,Y47) else cov57_4=0
if (cov(Y46,Y47)>0) cov67_4=cov(Y46,Y47) else cov67_4=0

cov4=c(cov12_4,cov13_4,cov14_4,cov15_4,cov16_4,cov17_4,cov23_4,cov24_4,
      cov25_4,cov26_4,cov27_4,cov34_4,cov35_4,cov36_4,cov37_4,cov45_4,
      cov46_4,cov47_4,cov56_4,cov57_4,cov67_4)

theta48=min(mean(cov4),(alfa4/(1+exp(-1*(delta4+gama4*t[1])))),
            (alfa4/(1+exp(-1*(delta4+gama4*t[2])))),
            (alfa4/(1+exp(-1*(delta4+gama4*t[3])))),
            (alfa4/(1+exp(-1*(delta4+gama4*t[4])))),
            (alfa4/(1+exp(-1*(delta4+gama4*t[5])))),
            (alfa4/(1+exp(-1*(delta4+gama4*t[6])))),
            (alfa4/(1+exp(-1*(delta4+gama4*t[7])))))
theta48_ini=theta48

q=ceiling((max(Y)-1)/2)
quad=gauss.quad(q,kind="laguerre")
xk=quad$nodes; Ak=quad$weights

p=c(alfa1_ini,delta1_ini,gama1_ini,theta18_ini,
    alfa2_ini,delta2_ini,gama2_ini,theta28_ini,
    alfa3_ini,delta3_ini,gama3_ini,theta38_ini,
    alfa4_ini,delta4_ini,gama4_ini,theta48_ini,sigma2b_ini)

#####Início do Processo Iterativo

```

```

nk=length(Yt1)
m2lnL=1000; m2lnlike=1000
m2lnL1=0; m2lnL2=0
iter=0
difllike1=1000; difllike2=1000
prec=1e-6; maxit=500

while ((difllike1 > prec) & (iter < maxit )) {
  iter=iter+1
  m2lnL1=m2lnL; m2lnL2=m2lnlike

  #####PASSO E#####/

  ##### Esperança condicional de (Wij8/Yij)
  ebdy=febdy(p); ew8dby=few8dby(p)
  ew18=ew8dby[,1]; ew28=ew8dby[,2]
  ew38=ew8dby[,3]; ew48=ew8dby[,4]

  #####PASSO M#####/
  ##### Maximização da função de verossimilhança
  out1<-optim(c(alfa1,delta1,gama1,theta18,alfa2,delta2,gama2,theta28,
               alfa3,delta3,gama3,theta38,alfa4,delta4,gama4,theta48,sigma2b),
             mlnv,grd,method="BFGS",hessian=TRUE)

  alfa1=out1$par[1];delta1=out1$par[2];gama1=out1$par[3];theta18=out1$par[4]
  alfa2=out1$par[5];delta2=out1$par[6];gama2=out1$par[7];theta28=out1$par[8]
  alfa3=out1$par[9];delta3=out1$par[10];gama3=out1$par[11];theta38=out1$par[12]
  alfa4=out1$par[13];delta4=out1$par[14];gama4=out1$par[15];theta48=out1$par[16]
  sigma2b=out1$par[17]

  p=c(alfa1,delta1,gama1,theta18,alfa2,delta2,gama2,theta28,
      alfa3,delta3,gama3,theta38,alfa4,delta4,gama4,theta48,sigma2b)

  converge=out1$convergence; names(converge)<-('converge')

  m2lnL=2*out1$value; m2lnlike=-2*lfy(p)
  difllike1=abs((m2lnL-m2lnL1)/m2lnL1); difllike2=abs((m2lnlike-m2lnL2)/m2lnL2)

  names(nsimul) <- ('Simul'); names(iter)<-('iter')
  printvectort1<-c(alfa1,delta1,gama1,theta18)
  printvectort2<-c(alfa2,delta2,gama2,theta28)
  printvectort3<-c(alfa3,delta3,gama3,theta38)
  printvectort4<-c(alfa4,delta4,gama4,theta48)
  printvectorsigma2b<-c(sigma2b)
  printvectorvero<-c(m2lnL,difllike1,m2lnlike,difllike2)

```

```

names(printvectort1)<-c('alfa1','delta1','gama1','theta18')
names(printvectort2)<-c('alfa2','delta2','gama2','theta28')
names(printvectort3)<-c('alfa3','delta3','gama3','theta38')
names(printvectort4)<-c('alfa4','delta4','gama4','theta48')
names(printvectorsigma2b)<-c('sigma2b')
names(printvectorvero)<-c('-2*logvero1','Rel.Dif.logvero1',
                          '-2*logvero2','Rel.Dif.logvero2')

print.default(nsimul); print.default(iter)
print.default(converge,digits=10)
print.default(printvectort1,digits=10)
print.default(printvectort2,digits=10)
print.default(printvectort3,digits=10)
print.default(printvectort4,digits=10)
print.default(printvectorsigma2b,digits=10)
print.default(printvectorvero,digits=10)
}
result[k,]=c(nsimul,iter,converge,m2lnL,difllike1,m2lnlike,difllike2,
             alfa1_ini,delta1_ini,gama1_ini,theta18_ini,
             alfa2_ini,delta2_ini,gama2_ini,theta28_ini,
             alfa3_ini,delta3_ini,gama3_ini,theta38_ini,
             alfa4_ini,delta4_ini,gama4_ini,theta48_ini,sigma2b_ini)

result[k+1,]=c(nsimul,iter,converge,m2lnL,difllike1,m2lnlike,difllike2,
              alfa1,delta1,gama1,theta18,alfa2,delta2,gama2,theta28,
              alfa3,delta3,gama3,theta38,alfa4,delta4,gama4,theta48,sigma2b)
k=k+2
nsimul=nsimul+1
}
)

```

B.4 - Delineamento completamente aleatorizado na presença de superdispersão

```

library(nlreg)

# PARÂMETROS DA CURVA LOGÍSTICA PARA O TRATAMENTO 1
alfa1p=359.916; delta1p=-3.769; gama1p=0.022
# PARÂMETROS DA CURVA LOGÍSTICA PARA O TRATAMENTO 2
alfa2p=391.870; delta2p=-3.441; gama2p=0.021
# PARÂMETROS DA CURVA LOGÍSTICA PARA O TRATAMENTO 3
alfa3p=374.896; delta3p=-3.536; gama3p=0.022
# PARÂMETROS DA CURVA LOGÍSTICA PARA O TRATAMENTO 4
alfa4p=334.128; delta4p=-3.981; gama4p=0.026

psip=0.10

```

```

#Cálculos dos valores sobre a curva em cada tempo
t=c(1,46,91,136,181,226,271)

media = function(alfa,delta,gama,t){
  mu=matrix(numeric(length(t)),ncol=1)
  for (i in 1:length(t)){
    mu[i]=alfa/(1+exp(-(delta+gama*t[i])))
  }
  mu
}

mu1=media(alfa1p,delta1p,gama1p,t); mu2=media(alfa2p,delta2p,gama2p,t)
mu3=media(alfa3p,delta3p,gama3p,t); mu4=media(alfa4p,delta4p,gama4p,t)

##### Parâmetros de Covariância #####
theta18p=8; theta28p=12; theta38p=10; theta48p=6

n=5
trat=c(rep(1,n),rep(2,n),rep(3,n),rep(4,n))
trat1=c(rep(1,n),rep(0,3*n))
trat2=c(rep(0,n),rep(1,n),rep(0,2*n))
trat3=c(rep(0,2*n),rep(1,n),rep(0,n))
trat4=c(rep(0,3*n),rep(1,n))

#### Cálculo do log da distribuição marginal de Yij #####
lfy <- function(p) {
fy=matrix(0,nrow=nk,ncol=1); lfy=matrix(0,nrow=nk,ncol=1)
for (i in 1:nk){
  for (w8 in 0:min(Yt1[i],Yt2[i],Yt3[i],Yt4[i],Yt5[i],Yt6[i],Yt7[i])){
    lintfy=(log((p[4])**w8)+
      (Yt1[i]-w8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[1]))))-p[4]))+
      (Yt2[i]-w8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[2]))))-p[4]))+
      (Yt3[i]-w8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[3]))))-p[4]))+
      (Yt4[i]-w8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[4]))))-p[4]))+
      (Yt5[i]-w8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[5]))))-p[4]))+
      (Yt6[i]-w8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[6]))))-p[4]))+
      (Yt7[i]-w8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[7]))))-p[4]))-
      (1/p[17])*log(p[17])-lgamma(1/p[17])+
      lgamma(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+(1/p[17]))-
      lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-lgamma(Yt3[i]-w8+1)-
      lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-lgamma(Yt6[i]-w8+1)-
      lgamma(Yt7[i]-w8+1)-(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-
      6*w8+(1/p[17]))*log((p[1]/(1+exp(-1*(p[2]+p[3]*t[1]))))+
      (p[1]/(1+exp(-1*(p[2]+p[3]*t[2]))))+
      (p[1]/(1+exp(-1*(p[2]+p[3]*t[3]))))+
      (p[1]/(1+exp(-1*(p[2]+p[3]*t[4]))))+
      (p[1]/(1+exp(-1*(p[2]+p[3]*t[5]))))+
      (p[1]/(1+exp(-1*(p[2]+p[3]*t[6]))))+
      (p[1]/(1+exp(-1*(p[2]+p[3]*t[7]))))
  }
}

```



```

(p[1]/(1+exp(-1*(p[2]+p[3]*t[7]))))-6*p[4]+(1/p[17])))*trat1[i]+
(log((p[8])**w8)+
(Yt1[i]-w8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[1]))))-p[8])))+
(Yt2[i]-w8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[2]))))-p[8])))+
(Yt3[i]-w8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[3]))))-p[8])))+
(Yt4[i]-w8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[4]))))-p[8])))+
(Yt5[i]-w8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[5]))))-p[8])))+
(Yt6[i]-w8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[6]))))-p[8])))+
(Yt7[i]-w8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[7]))))-p[8])))-
(1/p[17])*log(p[17])-lgamma(1/p[17])+
lgamma(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+(1/p[17]))-
lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-lgamma(Yt3[i]-w8+1)-
lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-lgamma(Yt6[i]-w8+1)-
lgamma(Yt7[i]-w8+1)-(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-
6*w8+(1/p[17]))*log((p[5]/(1+exp(-1*(p[6]+p[7]*t[1]))))+
(p[5]/(1+exp(-1*(p[6]+p[7]*t[2]))))+
(p[5]/(1+exp(-1*(p[6]+p[7]*t[3]))))+
(p[5]/(1+exp(-1*(p[6]+p[7]*t[4]))))+
(p[5]/(1+exp(-1*(p[6]+p[7]*t[5]))))+
(p[5]/(1+exp(-1*(p[6]+p[7]*t[6]))))+
(p[5]/(1+exp(-1*(p[6]+p[7]*t[7]))))-6*p[8]+(1/p[17])))*trat2[i]+
(log((p[12])**w8)+
(Yt1[i]-w8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[1]))))-p[12])))+
(Yt2[i]-w8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[2]))))-p[12])))+
(Yt3[i]-w8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[3]))))-p[12])))+
(Yt4[i]-w8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[4]))))-p[12])))+
(Yt5[i]-w8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[5]))))-p[12])))+
(Yt6[i]-w8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[6]))))-p[12])))+
(Yt7[i]-w8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[7]))))-p[12])))-
(1/p[17])*log(p[17])-lgamma(1/p[17])+
lgamma(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+(1/p[17]))-
lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-lgamma(Yt3[i]-w8+1)-
lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-lgamma(Yt6[i]-w8+1)-
lgamma(Yt7[i]-w8+1)-(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-
6*w8+(1/p[17]))*log((p[9]/(1+exp(-1*(p[10]+p[11]*t[1]))))+
(p[9]/(1+exp(-1*(p[10]+p[11]*t[2]))))+
(p[9]/(1+exp(-1*(p[10]+p[11]*t[3]))))+
(p[9]/(1+exp(-1*(p[10]+p[11]*t[4]))))+
(p[9]/(1+exp(-1*(p[10]+p[11]*t[5]))))+
(p[9]/(1+exp(-1*(p[10]+p[11]*t[6]))))+
(p[9]/(1+exp(-1*(p[10]+p[11]*t[7]))))-6*p[12]+(1/p[17])))*trat3[i]+
(log((p[16])**w8)+
(Yt1[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[1]))))-p[16])))+
(Yt2[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[2]))))-p[16])))+
(Yt3[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[3]))))-p[16])))+
(Yt4[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[4]))))-p[16])))+
(Yt5[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[5]))))-p[16])))+
(Yt6[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[6]))))-p[16])))+
(Yt7[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[7]))))-p[16])))-

```

```

(1/p[17])*log(p[17])-lgamma(1/p[17])+
  lgamma(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+(1/p[17]))-
  lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-lgamma(Yt3[i]-w8+1)-
  lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-lgamma(Yt6[i]-w8+1)-
  lgamma(Yt7[i]-w8+1)-(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-
6*w8+(1/p[17]))*log((p[13]/(1+exp(-1*(p[14]+p[15]*t[1]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[2]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[3]))))+ (p[13]/(1+exp(-1*(p[14]+p[15]*t[4]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[5]))))+ (p[13]/(1+exp(-1*(p[14]+p[15]*t[6]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[7]))))-6*p[16]+(1/p[17]))*trat4[i]
intfy=exp(lintfy)
fy[i]=fy[i]+intfy
lfy[i]=log(fy[i])
  }
}
sum(lfy)
}

```

Cálculo da esperança condicional de (Zij/Yij)

```

zesper <- function(p) {
nez=matrix(0,nrow=nk,ncol=1); dez=matrix(0,nrow=nk,ncol=1)
ez=matrix(0,nrow=nk,ncol=1)
for (i in 1:nk){
  for (w8 in 0:min(Yt1[i],Yt2[i],Yt3[i],Yt4[i],Yt5[i],Yt6[i],Yt7[i])){
    lint1=(log((p[4])**w8)+
(Yt1[i]-w8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[1]))))-p[4])))+
(Yt2[i]-w8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[2]))))-p[4])))+
(Yt3[i]-w8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[3]))))-p[4])))+
(Yt4[i]-w8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[4]))))-p[4])))+
(Yt5[i]-w8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[5]))))-p[4])))+
(Yt6[i]-w8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[6]))))-p[4])))+
(Yt7[i]-w8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[7]))))-p[4])))-
(1/p[17])*log(p[17])-lgamma(1/p[17])+
  lgamma(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+(1/p[17])+1)-
  lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-lgamma(Yt3[i]-w8+1)-
  lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-lgamma(Yt6[i]-w8+1)-
  lgamma(Yt7[i]-w8+1)-(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-
6*w8+(1/p[17])+1)*log((p[1]/(1+exp(-1*(p[2]+p[3]*t[1]))))+
(p[1]/(1+exp(-1*(p[2]+p[3]*t[2]))))+ (p[1]/(1+exp(-1*(p[2]+p[3]*t[3]))))+
(p[1]/(1+exp(-1*(p[2]+p[3]*t[4]))))+ (p[1]/(1+exp(-1*(p[2]+p[3]*t[5]))))+
(p[1]/(1+exp(-1*(p[2]+p[3]*t[6]))))+ (p[1]/(1+exp(-1*(p[2]+p[3]*t[7]))))-
6*p[4]+(1/p[17]))*trat1[i]+
(log((p[8])**w8)+
(Yt1[i]-w8)*log((((p[5]/(1+exp(-1*(p[6]+p[7]*t[1]))))-p[8])))+
(Yt2[i]-w8)*log((((p[5]/(1+exp(-1*(p[6]+p[7]*t[2]))))-p[8])))+

```

$$\begin{aligned}
& (Yt3[i]-w8)*\log((((p[5]/(1+\exp(-1*(p[6]+p[7]*t[3]))))-p[8])))+ \\
& (Yt4[i]-w8)*\log((((p[5]/(1+\exp(-1*(p[6]+p[7]*t[4]))))-p[8])))+ \\
& (Yt5[i]-w8)*\log((((p[5]/(1+\exp(-1*(p[6]+p[7]*t[5]))))-p[8])))+ \\
& (Yt6[i]-w8)*\log((((p[5]/(1+\exp(-1*(p[6]+p[7]*t[6]))))-p[8])))+ \\
& (Yt7[i]-w8)*\log((((p[5]/(1+\exp(-1*(p[6]+p[7]*t[7]))))-p[8])))- \\
& (1/p[17])*\log(p[17])-lgamma(1/p[17])+ \\
& \quad lgamma(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+(1/p[17])+1)- \\
& \quad lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-lgamma(Yt3[i]-w8+1)- \\
& \quad lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-lgamma(Yt6[i]-w8+1)- \\
& \quad lgamma(Yt7[i]-w8+1)-(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]- \\
& 6*w8+(1/p[17])+1)*\log((p[5]/(1+\exp(-1*(p[6]+p[7]*t[1]))))+ \\
& (p[5]/(1+\exp(-1*(p[6]+p[7]*t[2]))))+ (p[5]/(1+\exp(-1*(p[6]+p[7]*t[3]))))+ \\
& (p[5]/(1+\exp(-1*(p[6]+p[7]*t[4]))))+ (p[5]/(1+\exp(-1*(p[6]+p[7]*t[5]))))+ \\
& (p[5]/(1+\exp(-1*(p[6]+p[7]*t[6]))))+ (p[5]/(1+\exp(-1*(p[6]+p[7]*t[7]))))- \\
& 6*p[8]+(1/p[17])))*trat2[i]+ \\
& \log((p[12])**w8)+ \\
& (Yt1[i]-w8)*\log((((p[9]/(1+\exp(-1*(p[10]+p[11]*t[1]))))-p[12])))+ \\
& (Yt2[i]-w8)*\log((((p[9]/(1+\exp(-1*(p[10]+p[11]*t[2]))))-p[12])))+ \\
& (Yt3[i]-w8)*\log((((p[9]/(1+\exp(-1*(p[10]+p[11]*t[3]))))-p[12])))+ \\
& (Yt4[i]-w8)*\log((((p[9]/(1+\exp(-1*(p[10]+p[11]*t[4]))))-p[12])))+ \\
& (Yt5[i]-w8)*\log((((p[9]/(1+\exp(-1*(p[10]+p[11]*t[5]))))-p[12])))+ \\
& (Yt6[i]-w8)*\log((((p[9]/(1+\exp(-1*(p[10]+p[11]*t[6]))))-p[12])))+ \\
& (Yt7[i]-w8)*\log((((p[9]/(1+\exp(-1*(p[10]+p[11]*t[7]))))-p[12])))- \\
& (1/p[17])*\log(p[17])-lgamma(1/p[17])+ \\
& \quad lgamma(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+(1/p[17])+1)- \\
& \quad lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-lgamma(Yt3[i]-w8+1)- \\
& \quad lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-lgamma(Yt6[i]-w8+1)- \\
& \quad lgamma(Yt7[i]-w8+1)-(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]- \\
& 6*w8+(1/p[17])+1)*\log((p[9]/(1+\exp(-1*(p[10]+p[11]*t[1]))))+ \\
& (p[9]/(1+\exp(-1*(p[10]+p[11]*t[2]))))+ (p[9]/(1+\exp(-1*(p[10]+p[11]*t[3]))))+ \\
& (p[9]/(1+\exp(-1*(p[10]+p[11]*t[4]))))+ (p[9]/(1+\exp(-1*(p[10]+p[11]*t[5]))))+ \\
& (p[9]/(1+\exp(-1*(p[10]+p[11]*t[6]))))+ (p[9]/(1+\exp(-1*(p[10]+p[11]*t[7]))))- \\
& 6*p[12]+(1/p[17])))*trat3[i]+ \\
& \log((p[16])**w8)+ \\
& (Yt1[i]-w8)*\log((((p[13]/(1+\exp(-1*(p[14]+p[15]*t[1]))))-p[16])))+ \\
& (Yt2[i]-w8)*\log((((p[13]/(1+\exp(-1*(p[14]+p[15]*t[2]))))-p[16])))+ \\
& (Yt3[i]-w8)*\log((((p[13]/(1+\exp(-1*(p[14]+p[15]*t[3]))))-p[16])))+ \\
& (Yt4[i]-w8)*\log((((p[13]/(1+\exp(-1*(p[14]+p[15]*t[4]))))-p[16])))+ \\
& (Yt5[i]-w8)*\log((((p[13]/(1+\exp(-1*(p[14]+p[15]*t[5]))))-p[16])))+ \\
& (Yt6[i]-w8)*\log((((p[13]/(1+\exp(-1*(p[14]+p[15]*t[6]))))-p[16])))+ \\
& (Yt7[i]-w8)*\log((((p[13]/(1+\exp(-1*(p[14]+p[15]*t[7]))))-p[16])))- \\
& (1/p[17])*\log(p[17])-lgamma(1/p[17])+ \\
& \quad lgamma(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+(1/p[17])+1)- \\
& \quad lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-lgamma(Yt3[i]-w8+1)- \\
& \quad lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-lgamma(Yt6[i]-w8+1)-
\end{aligned}$$

```

lgamma(Yt7[i]-w8+1)-(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-
6*w8+(1/p[17]))+1)*log((p[13]/(1+exp(-1*(p[14]+p[15]*t[1]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[2]))))+ (p[13]/(1+exp(-1*(p[14]+p[15]*t[3]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[4]))))+ (p[13]/(1+exp(-1*(p[14]+p[15]*t[5]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[6]))))+ (p[13]/(1+exp(-1*(p[14]+p[15]*t[7]))))-
6*p[16]+(1/p[17])))*trat4[i]
    int1=exp(lint1)
    nez[i]=nez[i]+int1

lint2=(log((p[4])**w8)+
(Yt1[i]-w8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[1]))))-p[4]))+
(Yt2[i]-w8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[2]))))-p[4]))+
(Yt3[i]-w8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[3]))))-p[4]))+
(Yt4[i]-w8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[4]))))-p[4]))+
(Yt5[i]-w8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[5]))))-p[4]))+
(Yt6[i]-w8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[6]))))-p[4]))+
(Yt7[i]-w8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[7]))))-p[4]))-
(1/p[17])*log(p[17])-lgamma(1/p[17]))+
lgamma(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+(1/p[17]))-
lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-lgamma(Yt3[i]-w8+1)-
lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-lgamma(Yt6[i]-w8+1)-
lgamma(Yt7[i]-w8+1)-(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-
6*w8+(1/p[17])))*log((p[1]/(1+exp(-1*(p[2]+p[3]*t[1]))))+
(p[1]/(1+exp(-1*(p[2]+p[3]*t[2]))))+ (p[1]/(1+exp(-1*(p[2]+p[3]*t[3]))))+
(p[1]/(1+exp(-1*(p[2]+p[3]*t[4]))))+ (p[1]/(1+exp(-1*(p[2]+p[3]*t[5]))))+
(p[1]/(1+exp(-1*(p[2]+p[3]*t[6]))))+ (p[1]/(1+exp(-1*(p[2]+p[3]*t[7]))))-
6*p[4]+(1/p[17])))*trat1[i]+
(log((p[8])**w8)+
(Yt1[i]-w8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[1]))))-p[8]))+
(Yt2[i]-w8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[2]))))-p[8]))+
(Yt3[i]-w8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[3]))))-p[8]))+
(Yt4[i]-w8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[4]))))-p[8]))+
(Yt5[i]-w8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[5]))))-p[8]))+
(Yt6[i]-w8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[6]))))-p[8]))+
(Yt7[i]-w8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[7]))))-p[8]))-
(1/p[17])*log(p[17])-lgamma(1/p[17]))+
lgamma(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+(1/p[17]))-
lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-lgamma(Yt3[i]-w8+1)-
lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-lgamma(Yt6[i]-w8+1)-
lgamma(Yt7[i]-w8+1)-(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-
6*w8+(1/p[17])))*log((p[5]/(1+exp(-1*(p[6]+p[7]*t[1]))))+
(p[5]/(1+exp(-1*(p[6]+p[7]*t[2]))))+ (p[5]/(1+exp(-1*(p[6]+p[7]*t[3]))))+
(p[5]/(1+exp(-1*(p[6]+p[7]*t[4]))))+ (p[5]/(1+exp(-1*(p[6]+p[7]*t[5]))))+
(p[5]/(1+exp(-1*(p[6]+p[7]*t[6]))))+ (p[5]/(1+exp(-1*(p[6]+p[7]*t[7]))))-
6*p[8]+(1/p[17])))*trat2[i]+

```

```

(log((p[12])**w8)+
(Yt1[i]-w8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[1]))))-p[12]))) +
(Yt2[i]-w8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[2]))))-p[12]))) +
(Yt3[i]-w8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[3]))))-p[12]))) +
(Yt4[i]-w8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[4]))))-p[12]))) +
(Yt5[i]-w8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[5]))))-p[12]))) +
(Yt6[i]-w8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[6]))))-p[12]))) +
(Yt7[i]-w8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[7]))))-p[12]))) -
(1/p[17])*log(p[17])-lgamma(1/p[17]))+
lgamma(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+(1/p[17]))-
lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-lgamma(Yt3[i]-w8+1)-
lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-lgamma(Yt6[i]-w8+1)-
lgamma(Yt7[i]-w8+1)-(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-
6*w8+(1/p[17])))*log((p[9]/(1+exp(-1*(p[10]+p[11]*t[1]))))+
(p[9]/(1+exp(-1*(p[10]+p[11]*t[2]))))+ (p[9]/(1+exp(-1*(p[10]+p[11]*t[3]))))+
(p[9]/(1+exp(-1*(p[10]+p[11]*t[4]))))+ (p[9]/(1+exp(-1*(p[10]+p[11]*t[5]))))+
(p[9]/(1+exp(-1*(p[10]+p[11]*t[6]))))+ (p[9]/(1+exp(-1*(p[10]+p[11]*t[7]))))-
6*p[12]+(1/p[17])))*trat3[i]+
(log((p[16])**w8)+
(Yt1[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[1]))))-p[16]))) +
(Yt2[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[2]))))-p[16]))) +
(Yt3[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[3]))))-p[16]))) +
(Yt4[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[4]))))-p[16]))) +
(Yt5[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[5]))))-p[16]))) +
(Yt6[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[6]))))-p[16]))) +
(Yt7[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[7]))))-p[16]))) -
(1/p[17])*log(p[17])-lgamma(1/p[17]))+
lgamma(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+(1/p[17]))-
lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-lgamma(Yt3[i]-w8+1)-
lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-lgamma(Yt6[i]-w8+1)-
lgamma(Yt7[i]-w8+1)-(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-
6*w8+(1/p[17])))*log((p[13]/(1+exp(-1*(p[14]+p[15]*t[1]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[2]))))+ (p[13]/(1+exp(-1*(p[14]+p[15]*t[3]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[4]))))+ (p[13]/(1+exp(-1*(p[14]+p[15]*t[5]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[6]))))+ (p[13]/(1+exp(-1*(p[14]+p[15]*t[7]))))-
6*p[16]+(1/p[17])))*trat4[i]
        int2=exp(lint2)
        dez[i]=dez[i]+int2
    }
    ez[i]=nez[i]/dez[i]
}
ez
}

```

Cálculo da esperança condicional de (Wij8/Zij,Yij)

```

fw8dyz <- function(p) {
numer8=matrix(0,nrow=nk,ncol=1); denom=matrix(0,nrow=nk,ncol=1)
ew8=matrix(0,nrow=nk,ncol=1)
for (i in 1:nk){
  for (w8 in 0:min(Yt1[i],Yt2[i],Yt3[i],Yt4[i],Yt5[i],Yt6[i],Yt7[i])){
    lp=(-1*p[4]*Z[i])+log((p[4]*Z[i])**w8)+
    (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[1]))))-p[4])*Z[i])+
    (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[2]))))-p[4])*Z[i])+
    (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[3]))))-p[4])*Z[i])+
    (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[4]))))-p[4])*Z[i])+
    (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[5]))))-p[4])*Z[i])+
    (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[6]))))-p[4])*Z[i])+
    (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[7]))))-p[4])*Z[i])+
    (Yt1[i]-w8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[1]))))-p[4])*Z[i]))+
    (Yt2[i]-w8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[2]))))-p[4])*Z[i]))+
    (Yt3[i]-w8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[3]))))-p[4])*Z[i]))+
    (Yt4[i]-w8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[4]))))-p[4])*Z[i]))+
    (Yt5[i]-w8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[5]))))-p[4])*Z[i]))+
    (Yt6[i]-w8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[6]))))-p[4])*Z[i]))+
    (Yt7[i]-w8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[7]))))-p[4])*Z[i]))-
    lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-
    lgamma(Yt3[i]-w8+1)-lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-
    lgamma(Yt6[i]-w8+1)-lgamma(Yt7[i]-w8+1))*trat1[i]+
    ((-1*p[8]*Z[i])+log((p[8]*Z[i])**w8)+
    (-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[1]))))-p[8])*Z[i])+
    (-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[2]))))-p[8])*Z[i])+
    (-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[3]))))-p[8])*Z[i])+
    (-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[4]))))-p[8])*Z[i])+
    (-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[5]))))-p[8])*Z[i])+
    (-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[6]))))-p[8])*Z[i])+
    (-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[7]))))-p[8])*Z[i])+
    (Yt1[i]-w8)*log((((p[5]/(1+exp(-1*(p[6]+p[7]*t[1]))))-p[8])*Z[i]))+
    (Yt2[i]-w8)*log((((p[5]/(1+exp(-1*(p[6]+p[7]*t[2]))))-p[8])*Z[i]))+
    (Yt3[i]-w8)*log((((p[5]/(1+exp(-1*(p[6]+p[7]*t[3]))))-p[8])*Z[i]))+
    (Yt4[i]-w8)*log((((p[5]/(1+exp(-1*(p[6]+p[7]*t[4]))))-p[8])*Z[i]))+
    (Yt5[i]-w8)*log((((p[5]/(1+exp(-1*(p[6]+p[7]*t[5]))))-p[8])*Z[i]))+
    (Yt6[i]-w8)*log((((p[5]/(1+exp(-1*(p[6]+p[7]*t[6]))))-p[8])*Z[i]))+
    (Yt7[i]-w8)*log((((p[5]/(1+exp(-1*(p[6]+p[7]*t[7]))))-p[8])*Z[i]))-
    lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-
    lgamma(Yt3[i]-w8+1)-lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-
    lgamma(Yt6[i]-w8+1)-lgamma(Yt7[i]-w8+1))*trat2[i]+
    ((-1*p[12]*Z[i])+log((p[12]*Z[i])**w8)+
    (-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[1]))))-p[12])*Z[i])+
    (-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[2]))))-p[12])*Z[i])+
    (-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[3]))))-p[12])*Z[i])+

```



```

(-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[4]))))-p[12])*Z[i])+
(-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[5]))))-p[12])*Z[i])+
(-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[6]))))-p[12])*Z[i])+
(-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[7]))))-p[12])*Z[i])+
(Yt1[i]-w8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[1]))))-p[12])*Z[i]))+
(Yt2[i]-w8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[2]))))-p[12])*Z[i]))+
(Yt3[i]-w8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[3]))))-p[12])*Z[i]))+
(Yt4[i]-w8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[4]))))-p[12])*Z[i]))+
(Yt5[i]-w8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[5]))))-p[12])*Z[i]))+
(Yt6[i]-w8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[6]))))-p[12])*Z[i]))+
(Yt7[i]-w8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[7]))))-p[12])*Z[i]))-
lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-
lgamma(Yt3[i]-w8+1)-lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-
lgamma(Yt6[i]-w8+1)-lgamma(Yt7[i]-w8+1))*trat3[i]+
((-1*p[16]*Z[i])+log((p[16]*Z[i])**w8)+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[1]))))-p[16])*Z[i])+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[2]))))-p[16])*Z[i])+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[3]))))-p[16])*Z[i])+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[4]))))-p[16])*Z[i])+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[5]))))-p[16])*Z[i])+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[6]))))-p[16])*Z[i])+
(-1*((p[13]/(1+exp(-1*(p[14]+p[15]*t[7]))))-p[16])*Z[i])+
(Yt1[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[1]))))-p[16])*Z[i]))+
(Yt2[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[2]))))-p[16])*Z[i]))+
(Yt3[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[3]))))-p[16])*Z[i]))+
(Yt4[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[4]))))-p[16])*Z[i]))+
(Yt5[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[5]))))-p[16])*Z[i]))+
(Yt6[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[6]))))-p[16])*Z[i]))+
(Yt7[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[7]))))-p[16])*Z[i]))-
lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-
lgamma(Yt3[i]-w8+1)-lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-
lgamma(Yt6[i]-w8+1)-lgamma(Yt7[i]-w8+1))*trat4[i]

ep=exp(lp)
prod=w8*ep
numer8[i]=numer8[i]+prod
denom[i]=denom[i]+ep
    }
ew8[i]=numer8[i]/denom[i]
}

ew8
}

```

```

##### Função do -log(verossimilhança) considerando ew8 como
##### verdadeira realização de w8

```

```

mlnv <- function(p) {
lnvero=(log((p[4])**ew8)+
(Yt1-ew8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[1]))))-p[4])))+
(Yt2-ew8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[2]))))-p[4])))+
(Yt3-ew8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[3]))))-p[4])))+
(Yt4-ew8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[4]))))-p[4])))+
(Yt5-ew8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[5]))))-p[4])))+
(Yt6-ew8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[6]))))-p[4])))+
(Yt7-ew8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[7]))))-p[4])))-
(1/p[17])*log(p[17])-lgamma(1/p[17])+
lgamma(Yt1+Yt2+Yt3+Yt4+Yt5+Yt6+Yt7-6*ew8+(1/p[17]))-lgamma(ew8+1)-
lgamma(Yt1-ew8+1)-lgamma(Yt2-ew8+1)-lgamma(Yt3-ew8+1)-lgamma(Yt4-ew8+1)-
lgamma(Yt5-ew8+1)-lgamma(Yt6-ew8+1)-lgamma(Yt7-ew8+1)-
(Yt1+Yt2+Yt3+Yt4+Yt5+Yt6+Yt7-6*ew8+(1/p[17]))*
log((p[1]/(1+exp(-1*(p[2]+p[3]*t[1])))+(p[1]/(1+exp(-1*(p[2]+p[3]*t[2]))))+
(p[1]/(1+exp(-1*(p[2]+p[3]*t[3])))+(p[1]/(1+exp(-1*(p[2]+p[3]*t[4]))))+
(p[1]/(1+exp(-1*(p[2]+p[3]*t[5])))+(p[1]/(1+exp(-1*(p[2]+p[3]*t[6]))))+
(p[1]/(1+exp(-1*(p[2]+p[3]*t[7]))))-6*p[4]+(1/p[17])))*trat1+
(log((p[8])**ew8)+
(Yt1-ew8)*log((((p[5]/(1+exp(-1*(p[6]+p[7]*t[1]))))-p[8])))+
(Yt2-ew8)*log((((p[5]/(1+exp(-1*(p[6]+p[7]*t[2]))))-p[8])))+
(Yt3-ew8)*log((((p[5]/(1+exp(-1*(p[6]+p[7]*t[3]))))-p[8])))+
(Yt4-ew8)*log((((p[5]/(1+exp(-1*(p[6]+p[7]*t[4]))))-p[8])))+
(Yt5-ew8)*log((((p[5]/(1+exp(-1*(p[6]+p[7]*t[5]))))-p[8])))+
(Yt6-ew8)*log((((p[5]/(1+exp(-1*(p[6]+p[7]*t[6]))))-p[8])))+
(Yt7-ew8)*log((((p[5]/(1+exp(-1*(p[6]+p[7]*t[7]))))-p[8])))-
(1/p[17])*log(p[17])-lgamma(1/p[17])+
lgamma(Yt1+Yt2+Yt3+Yt4+Yt5+Yt6+Yt7-6*ew8+(1/p[17]))-lgamma(ew8+1)-
lgamma(Yt1-ew8+1)-lgamma(Yt2-ew8+1)-lgamma(Yt3-ew8+1)-lgamma(Yt4-ew8+1)-
lgamma(Yt5-ew8+1)-lgamma(Yt6-ew8+1)-lgamma(Yt7-ew8+1)-
(Yt1+Yt2+Yt3+Yt4+Yt5+Yt6+Yt7-6*ew8+(1/p[17]))*
log((p[5]/(1+exp(-1*(p[6]+p[7]*t[1])))+(p[5]/(1+exp(-1*(p[6]+p[7]*t[2]))))+
(p[5]/(1+exp(-1*(p[6]+p[7]*t[3])))+(p[5]/(1+exp(-1*(p[6]+p[7]*t[4]))))+
(p[5]/(1+exp(-1*(p[6]+p[7]*t[5])))+(p[5]/(1+exp(-1*(p[6]+p[7]*t[6]))))+
(p[5]/(1+exp(-1*(p[6]+p[7]*t[7]))))-6*p[8]+(1/p[17])))*trat2+
(log((p[12])**ew8)+
(Yt1-ew8)*log((((p[9]/(1+exp(-1*(p[10]+p[11]*t[1]))))-p[12])))+
(Yt2-ew8)*log((((p[9]/(1+exp(-1*(p[10]+p[11]*t[2]))))-p[12])))+
(Yt3-ew8)*log((((p[9]/(1+exp(-1*(p[10]+p[11]*t[3]))))-p[12])))+
(Yt4-ew8)*log((((p[9]/(1+exp(-1*(p[10]+p[11]*t[4]))))-p[12])))+
(Yt5-ew8)*log((((p[9]/(1+exp(-1*(p[10]+p[11]*t[5]))))-p[12])))+
(Yt6-ew8)*log((((p[9]/(1+exp(-1*(p[10]+p[11]*t[6]))))-p[12])))+
(Yt7-ew8)*log((((p[9]/(1+exp(-1*(p[10]+p[11]*t[7]))))-p[12])))-
(1/p[17])*log(p[17])-lgamma(1/p[17])+
lgamma(Yt1+Yt2+Yt3+Yt4+Yt5+Yt6+Yt7-6*ew8+(1/p[17]))-lgamma(ew8+1)-

```



```

lgamma(Yt1-ew8+1)-lgamma(Yt2-ew8+1)-lgamma(Yt3-ew8+1)-lgamma(Yt4-ew8+1)-
lgamma(Yt5-ew8+1)-lgamma(Yt6-ew8+1)-lgamma(Yt7-ew8+1)-
(Yt1+Yt2+Yt3+Yt4+Yt5+Yt6+Yt7-6*ew8+(1/p[17]))*
log((p[9]/(1+exp(-1*(p[10]+p[11]*t[1])))))+(p[9]/(1+exp(-1*(p[10]+p[11]*t[2]))))+
(p[9]/(1+exp(-1*(p[10]+p[11]*t[3])))))+(p[9]/(1+exp(-1*(p[10]+p[11]*t[4]))))+
(p[9]/(1+exp(-1*(p[10]+p[11]*t[5])))))+(p[9]/(1+exp(-1*(p[10]+p[11]*t[6]))))+
(p[9]/(1+exp(-1*(p[10]+p[11]*t[7]))))-6*p[12]+(1/p[17]))*trat3+
(log((p[16])**ew8)+
(Yt1-ew8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[1]))))-p[16])))+
(Yt2-ew8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[2]))))-p[16])))+
(Yt3-ew8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[3]))))-p[16])))+
(Yt4-ew8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[4]))))-p[16])))+
(Yt5-ew8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[5]))))-p[16])))+
(Yt6-ew8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[6]))))-p[16])))+
(Yt7-ew8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[7]))))-p[16])))-
(1/p[17])*log(p[17])-lgamma(1/p[17])+
lgamma(Yt1+Yt2+Yt3+Yt4+Yt5+Yt6+Yt7-6*ew8+(1/p[17]))-lgamma(ew8+1)-
lgamma(Yt1-ew8+1)-lgamma(Yt2-ew8+1)-lgamma(Yt3-ew8+1)-lgamma(Yt4-ew8+1)-
lgamma(Yt5-ew8+1)-lgamma(Yt6-ew8+1)-lgamma(Yt7-ew8+1)-
(Yt1+Yt2+Yt3+Yt4+Yt5+Yt6+Yt7-6*ew8+(1/p[17]))*
log((p[13]/(1+exp(-1*(p[14]+p[15]*t[1])))))+(p[13]/(1+exp(-1*(p[14]+p[15]*t[2]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[3])))))+(p[13]/(1+exp(-1*(p[14]+p[15]*t[4]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[5])))))+(p[13]/(1+exp(-1*(p[14]+p[15]*t[6]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[7]))))-6*p[16]+(1/p[17]))*trat4
  lnveroM=sum(lnvero)
  -lnveroM
}

```

```

##### Definição do gradiente da função
##### a ser minimizada (-log(verossimilhança))
##### Vetor com -derivada(log(vero)) para cada observação

```

```

grd <- deriv(
~((log((p4)**ew8)+
  (Yt1-ew8)*log(((p1/(1+exp(-1*(p2+p3*t1))))-p4)))+
  (Yt2-ew8)*log(((p1/(1+exp(-1*(p2+p3*t2))))-p4)))+
  (Yt3-ew8)*log(((p1/(1+exp(-1*(p2+p3*t3))))-p4)))+
  (Yt4-ew8)*log(((p1/(1+exp(-1*(p2+p3*t4))))-p4)))+
  (Yt5-ew8)*log(((p1/(1+exp(-1*(p2+p3*t5))))-p4)))+
  (Yt6-ew8)*log(((p1/(1+exp(-1*(p2+p3*t6))))-p4)))+
  (Yt7-ew8)*log(((p1/(1+exp(-1*(p2+p3*t7))))-p4)))-
  (1/p17)*log(p17)-lgamma(1/p17)+
lgamma(Yt1+Yt2+Yt3+Yt4+Yt5+Yt6+Yt7-6*ew8+(1/p17))-lgamma(ew8+1)-
lgamma(Yt1-ew8+1)-lgamma(Yt2-ew8+1)-lgamma(Yt3-ew8+1)-lgamma(Yt4-ew8+1)-
lgamma(Yt5-ew8+1)-lgamma(Yt6-ew8+1)-lgamma(Yt7-ew8+1)-
(Yt1+Yt2+Yt3+Yt4+Yt5+Yt6+Yt7-6*ew8+(1/p17))*

```

```

log((p1/(1+exp(-1*(p2+p3*t1))))+(p1/(1+exp(-1*(p2+p3*t2))))+
(p1/(1+exp(-1*(p2+p3*t3))))+(p1/(1+exp(-1*(p2+p3*t4))))+
(p1/(1+exp(-1*(p2+p3*t5))))+(p1/(1+exp(-1*(p2+p3*t6))))+
(p1/(1+exp(-1*(p2+p3*t7))))-6*p4+(1/p17))*trat1+
(log((p8)**ew8)+
(Yt1-ew8)*log(((p5/(1+exp(-1*(p6+p7*t1))))-p8)))+
(Yt2-ew8)*log(((p5/(1+exp(-1*(p6+p7*t2))))-p8)))+
(Yt3-ew8)*log(((p5/(1+exp(-1*(p6+p7*t3))))-p8)))+
(Yt4-ew8)*log(((p5/(1+exp(-1*(p6+p7*t4))))-p8)))+
(Yt5-ew8)*log(((p5/(1+exp(-1*(p6+p7*t5))))-p8)))+
(Yt6-ew8)*log(((p5/(1+exp(-1*(p6+p7*t6))))-p8)))+
(Yt7-ew8)*log(((p5/(1+exp(-1*(p6+p7*t7))))-p8))-
(1/p17)*log(p17)-lgamma(1/p17)+
lgamma(Yt1+Yt2+Yt3+Yt4+Yt5+Yt6+Yt7-6*ew8+(1/p17))-lgamma(ew8+1)-
lgamma(Yt1-ew8+1)-lgamma(Yt2-ew8+1)-lgamma(Yt3-ew8+1)-lgamma(Yt4-ew8+1)-
lgamma(Yt5-ew8+1)-lgamma(Yt6-ew8+1)-lgamma(Yt7-ew8+1)-
(Yt1+Yt2+Yt3+Yt4+Yt5+Yt6+Yt7-6*ew8+(1/p17))*
log((p5/(1+exp(-1*(p6+p7*t1))))+(p5/(1+exp(-1*(p6+p7*t2))))+
(p5/(1+exp(-1*(p6+p7*t3))))+(p5/(1+exp(-1*(p6+p7*t4))))+
(p5/(1+exp(-1*(p6+p7*t5))))+(p5/(1+exp(-1*(p6+p7*t6))))+
(p5/(1+exp(-1*(p6+p7*t7))))-6*p8+(1/p17))*trat2+
(log((p12)**ew8)+
(Yt1-ew8)*log(((p9/(1+exp(-1*(p10+p11*t1))))-p12)))+
(Yt2-ew8)*log(((p9/(1+exp(-1*(p10+p11*t2))))-p12)))+
(Yt3-ew8)*log(((p9/(1+exp(-1*(p10+p11*t3))))-p12)))+
(Yt4-ew8)*log(((p9/(1+exp(-1*(p10+p11*t4))))-p12)))+
(Yt5-ew8)*log(((p9/(1+exp(-1*(p10+p11*t5))))-p12)))+
(Yt6-ew8)*log(((p9/(1+exp(-1*(p10+p11*t6))))-p12)))+
(Yt7-ew8)*log(((p9/(1+exp(-1*(p10+p11*t7))))-p12))-
(1/p17)*log(p17)-lgamma(1/p17)+
lgamma(Yt1+Yt2+Yt3+Yt4+Yt5+Yt6+Yt7-6*ew8+(1/p17))-lgamma(ew8+1)-
lgamma(Yt1-ew8+1)-lgamma(Yt2-ew8+1)-lgamma(Yt3-ew8+1)-lgamma(Yt4-ew8+1)-
lgamma(Yt5-ew8+1)-lgamma(Yt6-ew8+1)-lgamma(Yt7-ew8+1)-
(Yt1+Yt2+Yt3+Yt4+Yt5+Yt6+Yt7-6*ew8+(1/p17))*
log((p9/(1+exp(-1*(p10+p11*t1))))+(p9/(1+exp(-1*(p10+p11*t2))))+
(p9/(1+exp(-1*(p10+p11*t3))))+(p9/(1+exp(-1*(p10+p11*t4))))+
(p9/(1+exp(-1*(p10+p11*t5))))+(p9/(1+exp(-1*(p10+p11*t6))))+
(p9/(1+exp(-1*(p10+p11*t7))))-6*p12+(1/p17))*trat3+
(log((p16)**ew8)+
(Yt1-ew8)*log(((p13/(1+exp(-1*(p14+p15*t1))))-p16)))+
(Yt2-ew8)*log(((p13/(1+exp(-1*(p14+p15*t2))))-p16)))+
(Yt3-ew8)*log(((p13/(1+exp(-1*(p14+p15*t3))))-p16)))+
(Yt4-ew8)*log(((p13/(1+exp(-1*(p14+p15*t4))))-p16)))+
(Yt5-ew8)*log(((p13/(1+exp(-1*(p14+p15*t5))))-p16)))+
(Yt6-ew8)*log(((p13/(1+exp(-1*(p14+p15*t6))))-p16)))+

```

```

(Yt7-ew8)*log((((p13/(1+exp(-1*(p14+p15*t7))))-p16))) -
(1/p17)*log(p17)-lgamma(1/p17)+
lgamma(Yt1+Yt2+Yt3+Yt4+Yt5+Yt6+Yt7-6*ew8+(1/p17))-lgamma(ew8+1)-
lgamma(Yt1-ew8+1)-lgamma(Yt2-ew8+1)-lgamma(Yt3-ew8+1)-lgamma(Yt4-ew8+1)-
lgamma(Yt5-ew8+1)-lgamma(Yt6-ew8+1)-lgamma(Yt7-ew8+1)-
(Yt1+Yt2+Yt3+Yt4+Yt5+Yt6+Yt7-6*ew8+(1/p17))*
log((p13/(1+exp(-1*(p14+p15*t1))))+(p13/(1+exp(-1*(p14+p15*t2))))+
(p13/(1+exp(-1*(p14+p15*t3))))+(p13/(1+exp(-1*(p14+p15*t4))))+
(p13/(1+exp(-1*(p14+p15*t5))))+(p13/(1+exp(-1*(p14+p15*t6))))+
(p13/(1+exp(-1*(p14+p15*t7))))-6*p16+(1/p17))*trat4),
c("p1", "p2", "p3", "p4", "p5", "p6", "p7", "p8", "p9", "p10", "p11", "p12", "p13",
  "p14", "p15", "p16", "p17"), function(p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11,
  p12,p13,p14,p15,p16,p17,t1,t2,t3,t4,t5,t6,t7) NULL)

sgrd <- function(p,Yt1,Yt2,Yt3,Yt4,Yt5,Yt6,Yt7,ew8) {
  p1=p[1]; p2=p[2]; p3=p[3]; p4=p[4]; p5=p[5]; p6=p[6]; p7=p[7]
  p8=p[8]; p9=p[9]; p10=p[10]; p11=p[11]; p12=p[12]; p13=p[13]
  p14=p[14]; p15=p[15]; p16=p[16]; p17=p[17]
  t1=t[1]; t2=t[2]; t3=t[3]; t4=t[4]; t5=t[5]; t6=t[6]; t7=t[7]
  colSums(attr(grd(p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11,p12,p13,p14,p15,
    p16,p17,t1,t2,t3,t4,t5,t6,t7),"gradient"))
}

##### Início do Processo de Simulação #####
W11=matrix(0,nrow=n,ncol=1); W12=matrix(0,nrow=n,ncol=1)
W13=matrix(0,nrow=n,ncol=1); W14=matrix(0,nrow=n,ncol=1)
W15=matrix(0,nrow=n,ncol=1); W16=matrix(0,nrow=n,ncol=1)
W17=matrix(0,nrow=n,ncol=1); W18=matrix(0,nrow=n,ncol=1)

W21=matrix(0,nrow=n,ncol=1); W22=matrix(0,nrow=n,ncol=1)
W23=matrix(0,nrow=n,ncol=1); W24=matrix(0,nrow=n,ncol=1)
W25=matrix(0,nrow=n,ncol=1); W26=matrix(0,nrow=n,ncol=1)
W27=matrix(0,nrow=n,ncol=1); W28=matrix(0,nrow=n,ncol=1)

W31=matrix(0,nrow=n,ncol=1); W32=matrix(0,nrow=n,ncol=1)
W33=matrix(0,nrow=n,ncol=1); W34=matrix(0,nrow=n,ncol=1)
W35=matrix(0,nrow=n,ncol=1); W36=matrix(0,nrow=n,ncol=1)
W37=matrix(0,nrow=n,ncol=1); W38=matrix(0,nrow=n,ncol=1)

W41=matrix(0,nrow=n,ncol=1); W42=matrix(0,nrow=n,ncol=1)
W43=matrix(0,nrow=n,ncol=1); W44=matrix(0,nrow=n,ncol=1)
W45=matrix(0,nrow=n,ncol=1); W46=matrix(0,nrow=n,ncol=1)
W47=matrix(0,nrow=n,ncol=1); W48=matrix(0,nrow=n,ncol=1)

set.seed(33); smt=runif(1000)

```

```

smt=round(smt,digits=5); smt=smt*100000
result=matrix(0,nrow=2000,ncol=24)
k=1; nsimul=1
system.time(
while (nsimul <= 1000){
set.seed(smt[nsimul])

##### Gerando os efeitos aleatórios (Zij) seguindo
##### distribuição Gama(1/psip,1/psip)

# Efeito aleatório para o tratamento 1
Z1=rgamma(n, shape=1/psip, scale=psip)
# Efeito aleatório para o tratamento 2
Z2=rgamma(n, shape=1/psip, scale=psip)
# Efeito aleatório para o tratamento 3
Z3=rgamma(n, shape=1/psip, scale=psip)
# Efeito aleatório para o tratamento 4
Z4=rgamma(n, shape=1/psip, scale=psip)

Z=c(Z1,Z2,Z3,Z4)

##### Gerando os valores das variáveis latentes
##### condicionais aos valores de Zij
for (i in 1:n) {
#Variáveis latentes para o tratamento 1
W11[i]=rpois(1,(mu1[1]-theta18p)*Z1[i])
W12[i]=rpois(1,(mu1[2]-theta18p)*Z1[i])
W13[i]=rpois(1,(mu1[3]-theta18p)*Z1[i])
W14[i]=rpois(1,(mu1[4]-theta18p)*Z1[i])
W15[i]=rpois(1,(mu1[5]-theta18p)*Z1[i])
W16[i]=rpois(1,(mu1[6]-theta18p)*Z1[i])
W17[i]=rpois(1,(mu1[7]-theta18p)*Z1[i])
W18[i]=rpois(1,theta18p*Z1[i])

#Variáveis latentes para o tratamento 2
W21[i]=rpois(1,(mu2[1]-theta28p)*Z2[i])
W22[i]=rpois(1,(mu2[2]-theta28p)*Z2[i])
W23[i]=rpois(1,(mu2[3]-theta28p)*Z2[i])
W24[i]=rpois(1,(mu2[4]-theta28p)*Z2[i])
W25[i]=rpois(1,(mu2[5]-theta28p)*Z2[i])
W26[i]=rpois(1,(mu2[6]-theta28p)*Z2[i])
W27[i]=rpois(1,(mu2[7]-theta28p)*Z2[i])
W28[i]=rpois(1,theta28p*Z2[i])

#Variáveis latentes para o tratamento 3

```

```

W31[i]=rpois(1,(mu3[1]-theta38p)*Z3[i])
W32[i]=rpois(1,(mu3[2]-theta38p)*Z3[i])
W33[i]=rpois(1,(mu3[3]-theta38p)*Z3[i])
W34[i]=rpois(1,(mu3[4]-theta38p)*Z3[i])
W35[i]=rpois(1,(mu3[5]-theta38p)*Z3[i])
W36[i]=rpois(1,(mu3[6]-theta38p)*Z3[i])
W37[i]=rpois(1,(mu3[7]-theta38p)*Z3[i])
W38[i]=rpois(1,theta38p*Z3[i])

#Variáveis latentes para o tratamento 4
W41[i]=rpois(1,(mu4[1]-theta48p)*Z4[i])
W42[i]=rpois(1,(mu4[2]-theta48p)*Z4[i])
W43[i]=rpois(1,(mu4[3]-theta48p)*Z4[i])
W44[i]=rpois(1,(mu4[4]-theta48p)*Z4[i])
W45[i]=rpois(1,(mu4[5]-theta48p)*Z4[i])
W46[i]=rpois(1,(mu4[6]-theta48p)*Z4[i])
W47[i]=rpois(1,(mu4[7]-theta48p)*Z4[i])
W48[i]=rpois(1,theta48p*Z4[i])
}

#### Gerando os valores observados condicionais aos valores de Zij
#Variáveis observadas para o tratamento 1
Y11=W11+W18; Y12=W12+W18; Y13=W13+W18; Y14=W14+W18
Y15=W15+W18; Y16=W16+W18; Y17=W17+W18

Y1=cbind(Y11,Y12,Y13,Y14,Y15,Y16,Y17)
Ytrat1=rbind(t(t(Y11)),t(t(Y12)),t(t(Y13)),t(t(Y14)),t(t(Y15)),
             t(t(Y16)),t(t(Y17)))

#Variáveis observadas para o tratamento 2
Y21=W21+W28; Y22=W22+W28; Y23=W23+W28; Y24=W24+W28
Y25=W25+W28; Y26=W26+W28; Y27=W27+W28

Y2=cbind(Y21,Y22,Y23,Y24,Y25,Y26,Y27)
Ytrat2=rbind(t(t(Y21)),t(t(Y22)),t(t(Y23)),t(t(Y24)),t(t(Y25)),
             t(t(Y26)),t(t(Y27)))

#Variáveis observadas para o tratamento 3
Y31=W31+W38; Y32=W32+W38; Y33=W33+W38; Y34=W34+W38
Y35=W35+W38; Y36=W36+W38; Y37=W37+W38

Y3=cbind(Y31,Y32,Y33,Y34,Y35,Y36,Y37)
Ytrat3=rbind(t(t(Y31)),t(t(Y32)),t(t(Y33)),t(t(Y34)),t(t(Y35)),
             t(t(Y36)),t(t(Y37)))

```

```

#Variáveis observadas para o tratamento 4
Y41=W41+W48; Y42=W42+W48; Y43=W43+W48; Y44=W44+W48
Y45=W45+W48; Y46=W46+W48; Y47=W47+W48

Y4=cbind(Y41,Y42,Y43,Y44,Y45,Y46,Y47)
Ytrat4=rbind(t(t(Y41)),t(t(Y42)),t(t(Y43)),t(t(Y44)),t(t(Y45)),
             t(t(Y46)),t(t(Y47)))

Y=rbind(Y1,Y2,Y3,Y4)
Yt1=Y[,1]; Yt2=Y[,2]; Yt3=Y[,3]; Yt4=Y[,4]; Yt5=Y[,5]; Yt6=Y[,6]; Yt7=Y[,7]

tempo=c(rep(t[1],n),rep(t[2],n),rep(t[3],n),rep(t[4],n),rep(t[5],n),
        rep(t[6],n),rep(t[7],n))
Ytrat=data.frame(tempo,Ytrat1,Ytrat2,Ytrat3,Ytrat4)

#Encontrando valores iniciais para os parâmetros da regressão
init1 <- nlreg( Ytrat1 ~ (alfa1/(1+exp(-1*(delta1+gama1*tempo))))),
              start = c(alfa1=200, delta1=-1, gama1=0.05), data = Ytrat)
init2 <- nlreg( Ytrat2 ~ (alfa2/(1+exp(-1*(delta2+gama2*tempo))))),
              start = c(alfa2=200, delta2=-1, gama2=0.05), data = Ytrat)
init3 <- nlreg( Ytrat3 ~ (alfa3/(1+exp(-1*(delta3+gama3*tempo))))),
              start = c(alfa3=200, delta3=-1, gama3=0.05), data = Ytrat)
init4 <- nlreg( Ytrat4 ~ (alfa4/(1+exp(-1*(delta4+gama4*tempo))))),
              start = c(alfa4=200, delta4=-1, gama4=0.05), data = Ytrat)

alfa1=init1$coef[1]; delta1=init1$coef[2]; gama1=init1$coef[3]
alfa1_ini=alfa1; delta1_ini=delta1; gama1_ini=gama1
alfa2=init2$coef[1]; delta2=init2$coef[2]; gama2=init2$coef[3]
alfa2_ini=alfa2; delta2_ini=delta2; gama2_ini=gama2
alfa3=init3$coef[1]; delta3=init3$coef[2]; gama3=init3$coef[3]
alfa3_ini=alfa3; delta3_ini=delta3; gama3_ini=gama3
alfa4=init4$coef[1]; delta4=init4$coef[2]; gama4=init4$coef[3]
alfa4_ini=alfa4; delta4_ini=delta4; gama4_ini=gama4

psi=var(Z); psi_ini=psi

#Encontrando os valores iniciais para os parâmetros de covariância
if (cov(Y11,Y12)>0) cov12_1=cov(Y11,Y12) else cov12_1=0
if (cov(Y11,Y13)>0) cov13_1=cov(Y11,Y13) else cov13_1=0
if (cov(Y11,Y14)>0) cov14_1=cov(Y11,Y14) else cov14_1=0
if (cov(Y11,Y15)>0) cov15_1=cov(Y11,Y15) else cov15_1=0
if (cov(Y11,Y16)>0) cov16_1=cov(Y11,Y16) else cov16_1=0
if (cov(Y11,Y17)>0) cov17_1=cov(Y11,Y17) else cov17_1=0
if (cov(Y12,Y13)>0) cov23_1=cov(Y12,Y13) else cov23_1=0
if (cov(Y12,Y14)>0) cov24_1=cov(Y12,Y14) else cov24_1=0

```

```

if (cov(Y12,Y15)>0) cov25_1=cov(Y12,Y15) else cov25_1=0
if (cov(Y12,Y16)>0) cov26_1=cov(Y12,Y16) else cov26_1=0
if (cov(Y12,Y17)>0) cov27_1=cov(Y12,Y17) else cov27_1=0
if (cov(Y13,Y14)>0) cov34_1=cov(Y13,Y14) else cov34_1=0
if (cov(Y13,Y15)>0) cov35_1=cov(Y13,Y15) else cov35_1=0
if (cov(Y13,Y16)>0) cov36_1=cov(Y13,Y16) else cov36_1=0
if (cov(Y13,Y17)>0) cov37_1=cov(Y13,Y17) else cov37_1=0
if (cov(Y14,Y15)>0) cov45_1=cov(Y14,Y15) else cov45_1=0
if (cov(Y14,Y16)>0) cov46_1=cov(Y14,Y16) else cov46_1=0
if (cov(Y14,Y17)>0) cov47_1=cov(Y14,Y17) else cov47_1=0
if (cov(Y15,Y16)>0) cov56_1=cov(Y15,Y16) else cov56_1=0
if (cov(Y15,Y17)>0) cov57_1=cov(Y15,Y17) else cov57_1=0
if (cov(Y16,Y17)>0) cov67_1=cov(Y16,Y17) else cov67_1=0

cov1=c(cov12_1,cov13_1,cov14_1,cov15_1,cov16_1,cov17_1,cov23_1,cov24_1,
      cov25_1,cov26_1,cov27_1,cov34_1,cov35_1,cov36_1,cov37_1,cov45_1,
      cov46_1,cov47_1,cov56_1,cov57_1,cov67_1)

theta18=min(mean(cov1),(alfa1/(1+exp(-1*(delta1+gama1*t[1]))))-0.00001,
            (alfa1/(1+exp(-1*(delta1+gama1*t[2]))))-0.00001,
            (alfa1/(1+exp(-1*(delta1+gama1*t[3]))))-0.00001,
            (alfa1/(1+exp(-1*(delta1+gama1*t[4]))))-0.00001,
            (alfa1/(1+exp(-1*(delta1+gama1*t[5]))))-0.00001,
            (alfa1/(1+exp(-1*(delta1+gama1*t[6]))))-0.00001,
            (alfa1/(1+exp(-1*(delta1+gama1*t[7]))))-0.00001)
theta18_ini=theta18

if (cov(Y21,Y22)>0) cov12_2=cov(Y21,Y22) else cov12_2=0
if (cov(Y21,Y23)>0) cov13_2=cov(Y21,Y23) else cov13_2=0
if (cov(Y21,Y24)>0) cov14_2=cov(Y21,Y24) else cov14_2=0
if (cov(Y21,Y25)>0) cov15_2=cov(Y21,Y25) else cov15_2=0
if (cov(Y21,Y26)>0) cov16_2=cov(Y21,Y26) else cov16_2=0
if (cov(Y21,Y27)>0) cov17_2=cov(Y21,Y27) else cov17_2=0
if (cov(Y22,Y23)>0) cov23_2=cov(Y22,Y23) else cov23_2=0
if (cov(Y22,Y24)>0) cov24_2=cov(Y22,Y24) else cov24_2=0
if (cov(Y22,Y25)>0) cov25_2=cov(Y22,Y25) else cov25_2=0
if (cov(Y22,Y26)>0) cov26_2=cov(Y22,Y26) else cov26_2=0
if (cov(Y22,Y27)>0) cov27_2=cov(Y22,Y27) else cov27_2=0
if (cov(Y23,Y24)>0) cov34_2=cov(Y23,Y24) else cov34_2=0
if (cov(Y23,Y25)>0) cov35_2=cov(Y23,Y25) else cov35_2=0
if (cov(Y23,Y26)>0) cov36_2=cov(Y23,Y26) else cov36_2=0
if (cov(Y23,Y27)>0) cov37_2=cov(Y23,Y27) else cov37_2=0
if (cov(Y24,Y25)>0) cov45_2=cov(Y24,Y25) else cov45_2=0
if (cov(Y24,Y26)>0) cov46_2=cov(Y24,Y26) else cov46_2=0
if (cov(Y24,Y27)>0) cov47_2=cov(Y24,Y27) else cov47_2=0

```



```

if (cov(Y25,Y26)>0) cov56_2=cov(Y25,Y26) else cov56_2=0
if (cov(Y25,Y27)>0) cov57_2=cov(Y25,Y27) else cov57_2=0
if (cov(Y26,Y27)>0) cov67_2=cov(Y26,Y27) else cov67_2=0

cov2=c(cov12_2,cov13_2,cov14_2,cov15_2,cov16_2,cov17_2,cov23_2,cov24_2,
      cov25_2,cov26_2,cov27_2,cov34_2,cov35_2,cov36_2,cov37_2,cov45_2,
      cov46_2,cov47_2,cov56_2,cov57_2,cov67_2)

theta28=min(mean(cov2),(alfa2/(1+exp(-1*(delta2+gama2*t[1]))))-0.00001,
            (alfa2/(1+exp(-1*(delta2+gama2*t[2]))))-0.00001,
            (alfa2/(1+exp(-1*(delta2+gama2*t[3]))))-0.00001,
            (alfa2/(1+exp(-1*(delta2+gama2*t[4]))))-0.00001,
            (alfa2/(1+exp(-1*(delta2+gama2*t[5]))))-0.00001,
            (alfa2/(1+exp(-1*(delta2+gama2*t[6]))))-0.00001,
            (alfa2/(1+exp(-1*(delta2+gama2*t[7]))))-0.00001)
theta28_ini=theta28

if (cov(Y31,Y32)>0) cov12_3=cov(Y31,Y32) else cov12_3=0
if (cov(Y31,Y33)>0) cov13_3=cov(Y31,Y33) else cov13_3=0
if (cov(Y31,Y34)>0) cov14_3=cov(Y31,Y34) else cov14_3=0
if (cov(Y31,Y35)>0) cov15_3=cov(Y31,Y35) else cov15_3=0
if (cov(Y31,Y36)>0) cov16_3=cov(Y31,Y36) else cov16_3=0
if (cov(Y31,Y37)>0) cov17_3=cov(Y31,Y37) else cov17_3=0
if (cov(Y32,Y33)>0) cov23_3=cov(Y32,Y33) else cov23_3=0
if (cov(Y32,Y34)>0) cov24_3=cov(Y32,Y34) else cov24_3=0
if (cov(Y32,Y35)>0) cov25_3=cov(Y32,Y35) else cov25_3=0
if (cov(Y32,Y36)>0) cov26_3=cov(Y32,Y36) else cov26_3=0
if (cov(Y32,Y37)>0) cov27_3=cov(Y32,Y37) else cov27_3=0
if (cov(Y33,Y34)>0) cov34_3=cov(Y33,Y34) else cov34_3=0
if (cov(Y33,Y35)>0) cov35_3=cov(Y33,Y35) else cov35_3=0
if (cov(Y33,Y36)>0) cov36_3=cov(Y33,Y36) else cov36_3=0
if (cov(Y33,Y37)>0) cov37_3=cov(Y33,Y37) else cov37_3=0
if (cov(Y34,Y35)>0) cov45_3=cov(Y34,Y35) else cov45_3=0
if (cov(Y34,Y36)>0) cov46_3=cov(Y34,Y36) else cov46_3=0
if (cov(Y34,Y37)>0) cov47_3=cov(Y34,Y37) else cov47_3=0
if (cov(Y35,Y36)>0) cov56_3=cov(Y35,Y36) else cov56_3=0
if (cov(Y35,Y37)>0) cov57_3=cov(Y35,Y37) else cov57_3=0
if (cov(Y36,Y37)>0) cov67_3=cov(Y36,Y37) else cov67_3=0

cov3=c(cov12_3,cov13_3,cov14_3,cov15_3,cov16_3,cov17_3,cov23_3,cov24_3,
      cov25_3,cov26_3,cov27_3,cov34_3,cov35_3,cov36_3,cov37_3,cov45_3,
      cov46_3,cov47_3,cov56_3,cov57_3,cov67_3)

theta38=min(mean(cov3),(alfa3/(1+exp(-1*(delta3+gama3*t[1]))))-0.00001,
            (alfa3/(1+exp(-1*(delta3+gama3*t[2]))))-0.00001,

```



```

(alfa3/(1+exp(-1*(delta3+gama3*t[3]))))-0.00001,
(alfa3/(1+exp(-1*(delta3+gama3*t[4]))))-0.00001,
(alfa3/(1+exp(-1*(delta3+gama3*t[5]))))-0.00001,
(alfa3/(1+exp(-1*(delta3+gama3*t[6]))))-0.00001,
(alfa3/(1+exp(-1*(delta3+gama3*t[7]))))-0.00001)
theta38_ini=theta38

if (cov(Y41,Y42)>0) cov12_4=cov(Y41,Y42) else cov12_4=0
if (cov(Y41,Y43)>0) cov13_4=cov(Y41,Y43) else cov13_4=0
if (cov(Y41,Y44)>0) cov14_4=cov(Y41,Y44) else cov14_4=0
if (cov(Y41,Y45)>0) cov15_4=cov(Y41,Y45) else cov15_4=0
if (cov(Y41,Y46)>0) cov16_4=cov(Y41,Y46) else cov16_4=0
if (cov(Y41,Y47)>0) cov17_4=cov(Y41,Y47) else cov17_4=0
if (cov(Y42,Y43)>0) cov23_4=cov(Y42,Y43) else cov23_4=0
if (cov(Y42,Y44)>0) cov24_4=cov(Y42,Y44) else cov24_4=0
if (cov(Y42,Y45)>0) cov25_4=cov(Y42,Y45) else cov25_4=0
if (cov(Y42,Y46)>0) cov26_4=cov(Y42,Y46) else cov26_4=0
if (cov(Y42,Y47)>0) cov27_4=cov(Y42,Y47) else cov27_4=0
if (cov(Y43,Y44)>0) cov34_4=cov(Y43,Y44) else cov34_4=0
if (cov(Y43,Y45)>0) cov35_4=cov(Y43,Y45) else cov35_4=0
if (cov(Y43,Y46)>0) cov36_4=cov(Y43,Y46) else cov36_4=0
if (cov(Y43,Y47)>0) cov37_4=cov(Y43,Y47) else cov37_4=0
if (cov(Y44,Y45)>0) cov45_4=cov(Y44,Y45) else cov45_4=0
if (cov(Y44,Y46)>0) cov46_4=cov(Y44,Y46) else cov46_4=0
if (cov(Y44,Y47)>0) cov47_4=cov(Y44,Y47) else cov47_4=0
if (cov(Y45,Y46)>0) cov56_4=cov(Y45,Y46) else cov56_4=0
if (cov(Y45,Y47)>0) cov57_4=cov(Y45,Y47) else cov57_4=0
if (cov(Y46,Y47)>0) cov67_4=cov(Y46,Y47) else cov67_4=0

cov4=c(cov12_4,cov13_4,cov14_4,cov15_4,cov16_4,cov17_4,cov23_4,cov24_4,
cov25_4,cov26_4,cov27_4,cov34_4,cov35_4,cov36_4,cov37_4,cov45_4,
cov46_4,cov47_4,cov56_4,cov57_4,cov67_4)

theta48=min(mean(cov4),(alfa4/(1+exp(-1*(delta4+gama4*t[1]))))-0.00001,
(alfa4/(1+exp(-1*(delta4+gama4*t[2]))))-0.00001,
(alfa4/(1+exp(-1*(delta4+gama4*t[3]))))-0.00001,
(alfa4/(1+exp(-1*(delta4+gama4*t[4]))))-0.00001,
(alfa4/(1+exp(-1*(delta4+gama4*t[5]))))-0.00001,
(alfa4/(1+exp(-1*(delta4+gama4*t[6]))))-0.00001,
(alfa4/(1+exp(-1*(delta4+gama4*t[7]))))-0.00001)
theta48_ini=theta48

p=c(alfa1_ini,delta1_ini,gama1_ini,theta18_ini,
alfa2_ini,delta2_ini,gama2_ini,theta28_ini,
alfa3_ini,delta3_ini,gama3_ini,theta38_ini,

```

```

    alfa4_ini,delta4_ini,gama4_ini,theta48_ini,psi_ini)

#####Início do Processo Iterativo

nk=length(Yt1)
m2lnL=1000; m2lnlike=1000
m2lnL1=0; m2lnL2=0
iter=0
difllike1=1000; difllike2=1000
prec=1e-6; maxit=500

while ((difllike1 > prec) & (iter < maxit )) {
  iter=iter+1
  m2lnL1=m2lnL; m2lnL2=m2lnlike

  #####PASSO E#####/

  ##### Esperança condicional de Zij/Yij
  Z=zesper(p)

  ##### Esperança condicional de (Wij8/Zij,Yij)
  ew8=fw8dyz(p)

  #####PASSO M#####/

  ##### Maximização da função de verossimilhança
  out1<-optim(c(alfa1,delta1,gama1,theta18,alfa2,delta2,gama2,theta28,
               alfa3,delta3,gama3,theta38,alfa4,delta4,gama4,theta48,psi),
             mlnv,sgrd,method="BFGS",hessian=TRUE)

  alfa1=out1$par[1];delta1=out1$par[2];gama1=out1$par[3];theta18=out1$par[4]
  alfa2=out1$par[5];delta2=out1$par[6];gama2=out1$par[7];theta28=out1$par[8]
  alfa3=out1$par[9];delta3=out1$par[10];gama3=out1$par[11];theta38=out1$par[12]
  alfa4=out1$par[13];delta4=out1$par[14];gama4=out1$par[15];theta48=out1$par[16]
  psi=out1$par[17]

  p=c(alfa1,delta1,gama1,theta18,alfa2,delta2,gama2,theta28,
      alfa3,delta3,gama3,theta38,alfa4,delta4,gama4,theta48,psi)

  converge=out1$convergence; names(converge)<-('converge')

  m2lnL=2*out1$value; m2lnlike=-2*lfy(p)
  difllike1=abs((m2lnL-m2lnL1)/m2lnL1); difllike2=abs((m2lnlike-m2lnL2)/m2lnL2)

  names(nsimul) <- ('Simul'); names(iter)<-('iter')

```

```

printvectort1<-c(alfa1,delta1,gama1,theta18)
printvectort2<-c(alfa2,delta2,gama2,theta28)
printvectort3<-c(alfa3,delta3,gama3,theta38)
printvectort4<-c(alfa4,delta4,gama4,theta48)
printvectorpsi<-c(psi)
printvectorvero<-c(m2lnL,difllike1,m2lnlike,difllike2)

names(printvectort1)<-c('alfa1','delta1','gama1','theta18')
names(printvectort2)<-c('alfa2','delta2','gama2','theta28')
names(printvectort3)<-c('alfa3','delta3','gama3','theta38')
names(printvectort4)<-c('alfa4','delta4','gama4','theta48')
names(printvectorpsi)<-c('psi')
names(printvectorvero)<-c('-2*logvero1','Rel.Dif.logvero1',
                          '-2*logvero2','Rel.Dif.logvero2')

print.default(nsimul); print.default(iter)
print.default(converge,digits=10)
print.default(printvectort1,digits=10)
print.default(printvectort2,digits=10)
print.default(printvectort3,digits=10)
print.default(printvectort4,digits=10)
print.default(printvectorpsi,digits=10)
print.default(printvectorvero,digits=10)
}
result[k,]=c(nsimul,iter,converge,m2lnL,difllike1,m2lnlike,difllike2,
             alfa1_ini,delta1_ini,gama1_ini,theta18_ini,
             alfa2_ini,delta2_ini,gama2_ini,theta28_ini,
             alfa3_ini,delta3_ini,gama3_ini,theta38_ini,
             alfa4_ini,delta4_ini,gama4_ini,theta48_ini,psi_ini)
result[k+1,]=c(nsimul,iter,converge,m2lnL,difllike1,m2lnlike,difllike2,
              alfa1,delta1,gama1,theta18,alfa2,delta2,gama2,theta28,
              alfa3,delta3,gama3,theta38,alfa4,delta4,gama4,theta48,psi)
k=k+2
nsimul=nsimul+1
}
)

```

B.5 - Delineamento em blocos casualizados com efeito fixo para blocos na presença de superdispersão

```
library(nlreg)

# PARÂMETROS DA CURVA LOGÍSTICA PARA O TRATAMENTO 1
alfa1p=359.916; delta1p=-3.769; gama1p=0.022
# PARÂMETROS DA CURVA LOGÍSTICA PARA O TRATAMENTO 2
alfa2p=391.870; delta2p=-3.441; gama2p=0.021
# PARÂMETROS DA CURVA LOGÍSTICA PARA O TRATAMENTO 3
alfa3p=374.896; delta3p=-3.536; gama3p=0.022
# PARÂMETROS DA CURVA LOGÍSTICA PARA O TRATAMENTO 4
alfa4p=334.128; delta4p=-3.981; gama4p=0.026

psip=0.10

#Cálculos dos valores sobre a curva em cada tempo
t=c(1,46,91,136,181,226,271)

# Como função+bi >= 0, então bi>=-função
# Calculando os valores das funções para os parâmetros especificados
# Depois serão estipulados os valores iniciais para bi

fmedia = function(alfa,delta,gama,t){
  fmu=matrix(numeric(length(t)),ncol=1)
  for (i in 1:length(t)){
    fmu[i]=(alfa/(1+exp(-(delta+gama*t[i]))))
  }
  fmu
}

fmu1=fmedia(alfa1p,delta1p,gama1p,t)
fmu2=fmedia(alfa2p,delta2p,gama2p,t)
fmu3=fmedia(alfa3p,delta3p,gama3p,t)
fmu4=fmedia(alfa4p,delta4p,gama4p,t)

# EFEITOS DE BLOCOS (Baseado nos resultados das médias acima)
beta1p=0; beta2p=0.46; beta3p=1.85; beta4p=5.06; beta5p=2.93
betap=c(beta1p,beta2p,beta3p,beta4p,beta5p)

# CURVA MÉDIA CONSIDERANDO O EFEITO DE BLOCOS
media = function(alfa,delta,gama,t,beta){
  mu=matrix(0,nrow=5,ncol=length(t))
  for (i in 1:5){
    for (j in 1:length(t)){
      mu[i,j]=alfa/(1+exp(-(delta+gama*t[j]))) + beta[i]
```

```

    }
  }
  mu
}

mu1=media(alfa1p,delta1p,gama1p,t,betap);
mu2=media(alfa2p,delta2p,gama2p,t,betap)
mu3=media(alfa3p,delta3p,gama3p,t,betap)
mu4=media(alfa4p,delta4p,gama4p,t,betap)

##### Parâmetros de Covariância #####
theta18p=8; theta28p=12; theta38p=10; theta48p=6

n=5
trat=c(rep(1,n),rep(2,n),rep(3,n),rep(4,n))
trat1=c(rep(1,n),rep(0,3*n))
trat2=c(rep(0,n),rep(1,n),rep(0,2*n))
trat3=c(rep(0,2*n),rep(1,n),rep(0,n))
trat4=c(rep(0,3*n),rep(1,n))
bloco=rep(1:n,4)
b2=rep(c(0,1,0,0,0),4)
b3=rep(c(0,0,1,0,0),4)
b4=rep(c(0,0,0,1,0),4)
b5=rep(c(0,0,0,0,1),4)

#### Cálculo do log da distribuição marginal de Yij #####
lvy <- function(p) {
fy=matrix(0,nrow=nk,ncol=1)
lfy=matrix(0,nrow=nk,ncol=1)
for (i in 1:nk){
  for (w8 in 0:min(Yt1[i],Yt2[i],Yt3[i],Yt4[i],Yt5[i],Yt6[i],Yt7[i]))){
    lintfy=(log((p[4]+0.00001)**w8)+
(Yt1[i]-w8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[1]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)))+
(Yt2[i]-w8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[2]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)))+
(Yt3[i]-w8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[3]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)))+
(Yt4[i]-w8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[4]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)))+
(Yt5[i]-w8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[5]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)))+
(Yt6[i]-w8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[6]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)))+
(Yt7[i]-w8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[7]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)))-

```

```

(1/p[21])*log(p[21))-lgamma(1/p[21])+
lgamma(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+(1/p[21]))-
lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-lgamma(Yt3[i]-w8+1)-
lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-lgamma(Yt6[i]-w8+1)-
lgamma(Yt7[i]-w8+1)-(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+
(1/p[21]))*log((p[1]/(1+exp(-1*(p[2]+p[3]*t[1])))))+
(p[1]/(1+exp(-1*(p[2]+p[3]*t[2])))))+(p[1]/(1+exp(-1*(p[2]+p[3]*t[3])))))+
(p[1]/(1+exp(-1*(p[2]+p[3]*t[4])))))+(p[1]/(1+exp(-1*(p[2]+p[3]*t[5])))))+
(p[1]/(1+exp(-1*(p[2]+p[3]*t[6])))))+(p[1]/(1+exp(-1*(p[2]+p[3]*t[7])))))+
7*(p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i])+0.00001-
6*p[4]+(1/p[21]))*trat1[i]+
(log((p[8]+0.00001)**w8)+
(Yt1[i]-w8)*log((((p[5]/(1+exp(-1*(p[6]+p[7]*t[1])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001))))+
(Yt2[i]-w8)*log((((p[5]/(1+exp(-1*(p[6]+p[7]*t[2])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001))))+
(Yt3[i]-w8)*log((((p[5]/(1+exp(-1*(p[6]+p[7]*t[3])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001))))+
(Yt4[i]-w8)*log((((p[5]/(1+exp(-1*(p[6]+p[7]*t[4])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001))))+
(Yt5[i]-w8)*log((((p[5]/(1+exp(-1*(p[6]+p[7]*t[5])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001))))+
(Yt6[i]-w8)*log((((p[5]/(1+exp(-1*(p[6]+p[7]*t[6])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001))))+
(Yt7[i]-w8)*log((((p[5]/(1+exp(-1*(p[6]+p[7]*t[7])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001))))-
(1/p[21])*log(p[21))-lgamma(1/p[21])+
lgamma(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+(1/p[21]))-
lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-lgamma(Yt3[i]-w8+1)-
lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-lgamma(Yt6[i]-w8+1)-
lgamma(Yt7[i]-w8+1)-(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+
(1/p[21]))*log((p[5]/(1+exp(-1*(p[6]+p[7]*t[1])))))+
(p[5]/(1+exp(-1*(p[6]+p[7]*t[2])))))+(p[5]/(1+exp(-1*(p[6]+p[7]*t[3])))))+
(p[5]/(1+exp(-1*(p[6]+p[7]*t[4])))))+(p[5]/(1+exp(-1*(p[6]+p[7]*t[5])))))+
(p[5]/(1+exp(-1*(p[6]+p[7]*t[6])))))+(p[5]/(1+exp(-1*(p[6]+p[7]*t[7])))))+
7*(p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i])+0.00001-
6*p[8]+(1/p[21]))*trat2[i]+
(log((p[12]+0.00001)**w8)+
(Yt1[i]-w8)*log((((p[9]/(1+exp(-1*(p[10]+p[11]*t[1])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[12]+0.00001))))+
(Yt2[i]-w8)*log((((p[9]/(1+exp(-1*(p[10]+p[11]*t[2])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[12]+0.00001))))+
(Yt3[i]-w8)*log((((p[9]/(1+exp(-1*(p[10]+p[11]*t[3])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[12]+0.00001))))+
(Yt4[i]-w8)*log((((p[9]/(1+exp(-1*(p[10]+p[11]*t[4])))))+

```

```

p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[12]+0.00001)))+
(Yt5[i]-w8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[5]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[12]+0.00001)))+
(Yt6[i]-w8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[6]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[12]+0.00001)))+
(Yt7[i]-w8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[7]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[12]+0.00001)))-
(1/p[21])*log(p[21])-lgamma(1/p[21]))+
lgamma(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+(1/p[21]))-
lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-lgamma(Yt3[i]-w8+1)-
lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-lgamma(Yt6[i]-w8+1)-
lgamma(Yt7[i]-w8+1)-(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+
(1/p[21]))*log((p[9]/(1+exp(-1*(p[10]+p[11]*t[1]))))+
(p[9]/(1+exp(-1*(p[10]+p[11]*t[2]))))+
(p[9]/(1+exp(-1*(p[10]+p[11]*t[3]))))+
(p[9]/(1+exp(-1*(p[10]+p[11]*t[4]))))+
(p[9]/(1+exp(-1*(p[10]+p[11]*t[5]))))+
(p[9]/(1+exp(-1*(p[10]+p[11]*t[6]))))+
(p[9]/(1+exp(-1*(p[10]+p[11]*t[7]))))+
7*(p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i])+0.00001-
6*p[12]+(1/p[21])))*trat3[i]+
(log((p[16]+0.00001)**w8)+
(Yt1[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[1]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[16]+0.00001)))+
(Yt2[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[2]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[16]+0.00001)))+
(Yt3[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[3]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[16]+0.00001)))+
(Yt4[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[4]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[16]+0.00001)))+
(Yt5[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[5]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[16]+0.00001)))+
(Yt6[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[6]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[16]+0.00001)))+
(Yt7[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[7]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[16]+0.00001)))-
(1/p[21])*log(p[21])-lgamma(1/p[21]))+
lgamma(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+(1/p[21]))-
lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-lgamma(Yt3[i]-w8+1)-
lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-lgamma(Yt6[i]-w8+1)-lgamma(Yt7[i]-w8+1)-
(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+
(1/p[21]))*log((p[13]/(1+exp(-1*(p[14]+p[15]*t[1]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[2]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[3]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[4]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[5]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[6]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[7]))))+
7*(p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i])+0.00001-
6*p[16]+(1/p[21])))*trat4[i]
intfy=exp(lintfy)

```

```

    fy[i]=fy[i]+intfy
    lfy[i]=log(fy[i])
  }
}
sum(lfy)
}

### Cálculo da esperança condicional de (Zij/Yij) #####
zesper <- function(p) {
  nez=matrix(0,nrow=nk,ncol=1)
  dez=matrix(0,nrow=nk,ncol=1)
  ez=matrix(0,nrow=nk,ncol=1)
  for (i in 1:nk){
    for (w8 in 0:min(Yt1[i],Yt2[i],Yt3[i],Yt4[i],Yt5[i],Yt6[i],Yt7[i]))){
      lint1=(log((p[4]+0.00001)**w8)+
        (Yt1[i]-w8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[1]))))+
        p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)))+
        (Yt2[i]-w8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[2]))))+
        p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)))+
        (Yt3[i]-w8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[3]))))+
        p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)))+
        (Yt4[i]-w8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[4]))))+
        p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)))+
        (Yt5[i]-w8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[5]))))+
        p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)))+
        (Yt6[i]-w8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[6]))))+
        p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)))+
        (Yt7[i]-w8)*log((((p[1]/(1+exp(-1*(p[2]+p[3]*t[7]))))+
        p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)))-
        (1/p[21])*log(p[21])-lgamma(1/p[21]))+
        lgamma(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+(1/p[21])+1)-
        lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-lgamma(Yt3[i]-w8+1)-
        lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-lgamma(Yt6[i]-w8+1)-
        lgamma(Yt7[i]-w8+1)-(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+
        (1/p[21])+1)*log((p[1]/(1+exp(-1*(p[2]+p[3]*t[1]))))+
        (p[1]/(1+exp(-1*(p[2]+p[3]*t[2]))))+
        (p[1]/(1+exp(-1*(p[2]+p[3]*t[3]))))+
        (p[1]/(1+exp(-1*(p[2]+p[3]*t[4]))))+
        (p[1]/(1+exp(-1*(p[2]+p[3]*t[5]))))+
        (p[1]/(1+exp(-1*(p[2]+p[3]*t[6]))))+
        (p[1]/(1+exp(-1*(p[2]+p[3]*t[7]))))+
        7*(p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i])+0.00001-
        6*p[4]+(1/p[21])))*trat1[i]+
        (log((p[8]+0.00001)**w8)+
        (Yt1[i]-w8)*log((((p[5]/(1+exp(-1*(p[6]+p[7]*t[1]))))+
        p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001)))+
        (Yt2[i]-w8)*log((((p[5]/(1+exp(-1*(p[6]+p[7]*t[2]))))+
        p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001)))+

```


$$\begin{aligned}
& (Yt3[i]-w8)*\log((((p[5]/(1+\exp(-1*(p[6]+p[7]*t[3])))))+ \\
& \quad p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001))))+ \\
& (Yt4[i]-w8)*\log((((p[5]/(1+\exp(-1*(p[6]+p[7]*t[4])))))+ \\
& \quad p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001))))+ \\
& (Yt5[i]-w8)*\log((((p[5]/(1+\exp(-1*(p[6]+p[7]*t[5])))))+ \\
& \quad p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001))))+ \\
& (Yt6[i]-w8)*\log((((p[5]/(1+\exp(-1*(p[6]+p[7]*t[6])))))+ \\
& \quad p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001))))+ \\
& (Yt7[i]-w8)*\log((((p[5]/(1+\exp(-1*(p[6]+p[7]*t[7])))))+ \\
& \quad p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001))))- \\
& (1/p[21])*\log(p[21])-lgamma(1/p[21])+ \\
& \quad lgamma(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+(1/p[21])+1)- \\
& \quad lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-lgamma(Yt3[i]-w8+1)- \\
& \quad lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-lgamma(Yt6[i]-w8+1)- \\
& \quad lgamma(Yt7[i]-w8+1)-(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+ \\
& \quad (1/p[21])+1)*\log((p[5]/(1+\exp(-1*(p[6]+p[7]*t[1])))))+ \\
& \quad (p[5]/(1+\exp(-1*(p[6]+p[7]*t[2])))))+(p[5]/(1+\exp(-1*(p[6]+p[7]*t[3])))))+ \\
& \quad (p[5]/(1+\exp(-1*(p[6]+p[7]*t[4])))))+(p[5]/(1+\exp(-1*(p[6]+p[7]*t[5])))))+ \\
& \quad (p[5]/(1+\exp(-1*(p[6]+p[7]*t[6])))))+(p[5]/(1+\exp(-1*(p[6]+p[7]*t[7])))))+ \\
& \quad 7*(p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i])+0.00001- \\
& \quad 6*p[8]+(1/p[21])))*trat2[i]+ \\
& (\log((p[12]+0.00001)**w8)+ \\
& (Yt1[i]-w8)*\log((((p[9]/(1+\exp(-1*(p[10]+p[11]*t[1])))))+ \\
& \quad p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[12]+0.00001))))+ \\
& (Yt2[i]-w8)*\log((((p[9]/(1+\exp(-1*(p[10]+p[11]*t[2])))))+ \\
& \quad p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[12]+0.00001))))+ \\
& (Yt3[i]-w8)*\log((((p[9]/(1+\exp(-1*(p[10]+p[11]*t[3])))))+ \\
& \quad p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[12]+0.00001))))+ \\
& (Yt4[i]-w8)*\log((((p[9]/(1+\exp(-1*(p[10]+p[11]*t[4])))))+ \\
& \quad p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[12]+0.00001))))+ \\
& (Yt5[i]-w8)*\log((((p[9]/(1+\exp(-1*(p[10]+p[11]*t[5])))))+ \\
& \quad p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[12]+0.00001))))+ \\
& (Yt6[i]-w8)*\log((((p[9]/(1+\exp(-1*(p[10]+p[11]*t[6])))))+ \\
& \quad p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[12]+0.00001))))+ \\
& (Yt7[i]-w8)*\log((((p[9]/(1+\exp(-1*(p[10]+p[11]*t[7])))))+ \\
& \quad p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[12]+0.00001))))- \\
& (1/p[21])*\log(p[21])-lgamma(1/p[21])+ \\
& \quad lgamma(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+(1/p[21])+1)- \\
& \quad lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-lgamma(Yt3[i]-w8+1)- \\
& \quad lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-lgamma(Yt6[i]-w8+1)-lgamma(Yt7[i]-w8+1)- \\
& \quad (Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+ \\
& \quad (1/p[21])+1)*\log((p[9]/(1+\exp(-1*(p[10]+p[11]*t[1])))))+ \\
& \quad (p[9]/(1+\exp(-1*(p[10]+p[11]*t[2])))))+(p[9]/(1+\exp(-1*(p[10]+p[11]*t[3])))))+ \\
& \quad (p[9]/(1+\exp(-1*(p[10]+p[11]*t[4])))))+(p[9]/(1+\exp(-1*(p[10]+p[11]*t[5])))))+ \\
& \quad (p[9]/(1+\exp(-1*(p[10]+p[11]*t[6])))))+(p[9]/(1+\exp(-1*(p[10]+p[11]*t[7])))))+
\end{aligned}$$

```

7*(p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i])+0.00001-
6*p[12]+(1/p[21])))*trat3[i]+
(log((p[16]+0.00001)**w8)+
(Yt1[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[1]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[16]+0.00001)))+
(Yt2[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[2]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[16]+0.00001)))+
(Yt3[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[3]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[16]+0.00001)))+
(Yt4[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[4]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[16]+0.00001)))+
(Yt5[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[5]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[16]+0.00001)))+
(Yt6[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[6]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[16]+0.00001)))+
(Yt7[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[7]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[16]+0.00001)))-
(1/p[21])*log(p[21])-lgamma(1/p[21])+
lgamma(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+(1/p[21])+1)-
lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-lgamma(Yt3[i]-w8+1)-
lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-lgamma(Yt6[i]-w8+1)-
lgamma(Yt7[i]-w8+1)-(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+
(1/p[21])+1)*log((p[13]/(1+exp(-1*(p[14]+p[15]*t[1]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[2]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[3]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[4]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[5]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[6]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[7]))))+
7*(p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i])+0.00001-
6*p[16]+(1/p[21])))*trat4[i]
int1=exp(lint1)
nez[i]=nez[i]+int1

```

```

lint2=(log((p[4]+0.00001)**w8)+
(Yt1[i]-w8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[1]))))+p
[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)))+
(Yt2[i]-w8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[2]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)))+
(Yt3[i]-w8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[3]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)))+
(Yt4[i]-w8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[4]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)))+
(Yt5[i]-w8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[5]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)))+
(Yt6[i]-w8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[6]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)))+
(Yt7[i]-w8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[7]))))+

```

```

p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001))) -
(1/p[21])*log(p[21])-lgamma(1/p[21])+
lgamma(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+(1/p[21]))-
lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-lgamma(Yt3[i]-w8+1)-
lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-lgamma(Yt6[i]-w8+1)-
lgamma(Yt7[i]-w8+1)-(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+
(1/p[21]))*log((p[1]/(1+exp(-1*(p[2]+p[3]*t[1])))))+
(p[1]/(1+exp(-1*(p[2]+p[3]*t[2])))))+(p[1]/(1+exp(-1*(p[2]+p[3]*t[3])))))+
(p[1]/(1+exp(-1*(p[2]+p[3]*t[4])))))+(p[1]/(1+exp(-1*(p[2]+p[3]*t[5])))))+
(p[1]/(1+exp(-1*(p[2]+p[3]*t[6])))))+(p[1]/(1+exp(-1*(p[2]+p[3]*t[7])))))+
7*(p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i])+0.00001-
6*p[4]+(1/p[21]))*trat1[i]+
(log((p[8]+0.00001)**w8)+
(Yt1[i]-w8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[1])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001))) +
(Yt2[i]-w8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[2])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001))) +
(Yt3[i]-w8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[3])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001))) +
(Yt4[i]-w8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[4])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001))) +
(Yt5[i]-w8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[5])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001))) +
(Yt6[i]-w8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[6])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001))) +
(Yt7[i]-w8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[7])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001))) -
(1/p[21])*log(p[21])-lgamma(1/p[21])+
lgamma(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+(1/p[21]))-
lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-lgamma(Yt3[i]-w8+1)-
lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-lgamma(Yt6[i]-w8+1)-
lgamma(Yt7[i]-w8+1)-(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+
(1/p[21]))*log((p[5]/(1+exp(-1*(p[6]+p[7]*t[1])))))+
(p[5]/(1+exp(-1*(p[6]+p[7]*t[2])))))+(p[5]/(1+exp(-1*(p[6]+p[7]*t[3])))))+
(p[5]/(1+exp(-1*(p[6]+p[7]*t[4])))))+(p[5]/(1+exp(-1*(p[6]+p[7]*t[5])))))+
(p[5]/(1+exp(-1*(p[6]+p[7]*t[6])))))+(p[5]/(1+exp(-1*(p[6]+p[7]*t[7])))))+
7*(p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i])+0.00001-
6*p[8]+(1/p[21]))*trat2[i]+
(log((p[12]+0.00001)**w8)+
(Yt1[i]-w8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[1])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[12]+0.00001))) +
(Yt2[i]-w8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[2])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[12]+0.00001))) +
(Yt3[i]-w8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[3])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[12]+0.00001))) +

```

```

(Yt4[i]-w8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[4]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[12]+0.00001)))+
(Yt5[i]-w8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[5]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[12]+0.00001)))+
(Yt6[i]-w8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[6]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[12]+0.00001)))+
(Yt7[i]-w8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[7]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[12]+0.00001)))-
(1/p[21])*log(p[21])-lgamma(1/p[21])+
lgamma(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+(1/p[21]))-
lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-lgamma(Yt3[i]-w8+1)-
lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-lgamma(Yt6[i]-w8+1)-
lgamma(Yt7[i]-w8+1)-(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+
(1/p[21]))*log((p[9]/(1+exp(-1*(p[10]+p[11]*t[1]))))+
(p[9]/(1+exp(-1*(p[10]+p[11]*t[2]))))+
(p[9]/(1+exp(-1*(p[10]+p[11]*t[3]))))+
(p[9]/(1+exp(-1*(p[10]+p[11]*t[4]))))+
(p[9]/(1+exp(-1*(p[10]+p[11]*t[5]))))+
(p[9]/(1+exp(-1*(p[10]+p[11]*t[6]))))+
(p[9]/(1+exp(-1*(p[10]+p[11]*t[7]))))+
7*(p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i])+0.00001-
6*p[12]+(1/p[21])))*trat3[i]+
(log((p[16]+0.00001)**w8)+
(Yt1[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[1]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[16]+0.00001)))+
(Yt2[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[2]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[16]+0.00001)))+
(Yt3[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[3]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[16]+0.00001)))+
(Yt4[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[4]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[16]+0.00001)))+
(Yt5[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[5]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[16]+0.00001)))+
(Yt6[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[6]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[16]+0.00001)))+
(Yt7[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[7]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[16]+0.00001)))-
(1/p[21])*log(p[21])-lgamma(1/p[21])+
lgamma(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+(1/p[21]))-
lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-lgamma(Yt3[i]-w8+1)-
lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-lgamma(Yt6[i]-w8+1)-
lgamma(Yt7[i]-w8+1)-(Yt1[i]+Yt2[i]+Yt3[i]+Yt4[i]+Yt5[i]+Yt6[i]+Yt7[i]-6*w8+
(1/p[21]))*log((p[13]/(1+exp(-1*(p[14]+p[15]*t[1]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[2]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[3]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[4]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[5]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[6]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[7]))))+
7*(p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i])+0.00001-
6*p[16]+(1/p[21])))*trat4[i]

```

```

        int2=exp(lint2)
        dez[i]=dez[i]+int2
    }
    ez[i]=nez[i]/dez[i]
}

ez

}

### Cálculo da esperança condicional de (Wij8/Zij,Yij) #####
fw8dyz <- function(p) {
  numer8=matrix(0,nrow=nk,ncol=1)
  denom=matrix(0,nrow=nk,ncol=1)
  ew8=matrix(0,nrow=nk,ncol=1)
  for (i in 1:nk){
    for (w8 in 0:min(Yt1[i],Yt2[i],Yt3[i],Yt4[i],Yt5[i],Yt6[i],Yt7[i])){
      lp=((-1*(p[4]+0.00001)*Z[i])+log(((p[4]+0.00001)*Z[i])**w8)+
        (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[1])))))+
        p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)*Z[i])+
        (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[2])))))+
        p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)*Z[i])+
        (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[3])))))+
        p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)*Z[i])+
        (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[4])))))+
        p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)*Z[i])+
        (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[5])))))+
        p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)*Z[i])+
        (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[6])))))+
        p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)*Z[i])+
        (-1*((p[1]/(1+exp(-1*(p[2]+p[3]*t[7])))))+
        p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)*Z[i])+
        (Yt1[i]-w8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[1])))))+
        p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)*Z[i]))+
        (Yt2[i]-w8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[2])))))+
        p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)*Z[i]))+
        (Yt3[i]-w8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[3])))))+
        p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)*Z[i]))+
        (Yt4[i]-w8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[4])))))+
        p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)*Z[i]))+
        (Yt5[i]-w8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[5])))))+
        p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)*Z[i]))+
        (Yt6[i]-w8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[6])))))+
        p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)*Z[i]))+
        (Yt7[i]-w8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[7])))))+
        p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[4]+0.00001)*Z[i]))-
      lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-

```

```

lgamma(Yt3[i]-w8+1)-lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-
lgamma(Yt6[i]-w8+1)-lgamma(Yt7[i]-w8+1))*trat1[i]+
((-1*(p[8]+0.00001)*Z[i])+log(((p[8]+0.00001)*Z[i])**w8)+
(-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[1])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001)*Z[i]))+
(-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[2])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001)*Z[i]))+
(-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[3])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001)*Z[i]))+
(-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[4])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001)*Z[i]))+
(-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[5])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001)*Z[i]))+
(-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[6])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001)*Z[i]))+
(-1*((p[5]/(1+exp(-1*(p[6]+p[7]*t[7])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001)*Z[i]))+
(Yt1[i]-w8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[1])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001)*Z[i]))+
(Yt2[i]-w8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[2])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001)*Z[i]))+
(Yt3[i]-w8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[3])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001)*Z[i]))+
(Yt4[i]-w8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[4])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001)*Z[i]))+
(Yt5[i]-w8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[5])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001)*Z[i]))+
(Yt6[i]-w8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[6])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001)*Z[i]))+
(Yt7[i]-w8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[7])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[8]+0.00001)*Z[i]))-
lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-
lgamma(Yt3[i]-w8+1)-lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-
lgamma(Yt6[i]-w8+1)-lgamma(Yt7[i]-w8+1))*trat2[i]+
((-1*(p[12]+0.00001)*Z[i])+log(((p[12]+0.00001)*Z[i])**w8)+
(-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[1])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[12]+0.00001)*Z[i]))+
(-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[2])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[12]+0.00001)*Z[i]))+
(-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[3])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[12]+0.00001)*Z[i]))+
(-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[4])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[12]+0.00001)*Z[i]))+
(-1*((p[9]/(1+exp(-1*(p[10]+p[11]*t[5])))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[12]+0.00001)*Z[i]))+

```



```

p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[16]+0.00001)*Z[i]))+
(Yt6[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[6]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[16]+0.00001)*Z[i]))+
(Yt7[i]-w8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[7]))))+
p[17]*b2[i]+p[18]*b3[i]+p[19]*b4[i]+p[20]*b5[i]-p[16]+0.00001)*Z[i]))-
lgamma(w8+1)-lgamma(Yt1[i]-w8+1)-lgamma(Yt2[i]-w8+1)-
lgamma(Yt3[i]-w8+1)-lgamma(Yt4[i]-w8+1)-lgamma(Yt5[i]-w8+1)-
lgamma(Yt6[i]-w8+1)-lgamma(Yt7[i]-w8+1))*trat4[i]

```

```

        ep=exp(lp)
        prod=w8*ep
        numer8[i]=numer8[i]+prod
        denom[i]=denom[i]+ep
    }
    ew8[i]=numer8[i]/denom[i]
}
ew8
}

```

```

##### Função do -log(verossimilhança)
##### considerando ew8 como verdadeira realização de w8
mlnv <- function(p) {
  lnvero=(log((p[4]+0.00001)**ew8)+
  (Yt1-ew8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[1]))))+
  p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5-p[4]+ 0.00001)))+
  (Yt2-ew8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[2]))))+
  p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5-p[4]+0.00001)))+
  (Yt3-ew8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[3]))))+
  p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5-p[4]+0.00001)))+
  (Yt4-ew8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[4]))))+
  p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5-p[4]+0.00001)))+
  (Yt5-ew8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[5]))))+
  p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5-p[4]+0.00001)))+
  (Yt6-ew8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[6]))))+
  p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5-p[4]+0.00001)))+
  (Yt7-ew8)*log(((p[1]/(1+exp(-1*(p[2]+p[3]*t[7]))))+
  p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5-p[4]+0.00001)))-
  (1/p[21])*log(p[21])-lgamma(1/p[21]))+
  lgamma(Yt1+Yt2+Yt3+Yt4+Yt5+Yt6+Yt7-6*ew8+(1/p[21]))-lgamma(ew8+1)-
  lgamma(Yt1-ew8+1)-lgamma(Yt2-ew8+1)-lgamma(Yt3-ew8+1)-
  lgamma(Yt4-ew8+1)-lgamma(Yt5-ew8+1)-lgamma(Yt6-ew8+1)-
  lgamma(Yt7-ew8+1)-(Yt1+Yt2+Yt3+Yt4+Yt5+Yt6+Yt7-6*ew8+(1/p[21]))*
  log((p[1]/(1+exp(-1*(p[2]+p[3]*t[1]))))+((p[1]/(1+exp(-1*(p[2]+p[3]*t[2]))))+
  (p[1]/(1+exp(-1*(p[2]+p[3]*t[3]))))+((p[1]/(1+exp(-1*(p[2]+p[3]*t[4]))))+
  (p[1]/(1+exp(-1*(p[2]+p[3]*t[5]))))+((p[1]/(1+exp(-1*(p[2]+p[3]*t[6]))))+

```



```

(p[1]/(1+exp(-1*(p[2]+p[3]*t[7]))))+
7*(p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5)+0.00001-
6*p[4]+(1/p[21])))*trat1+
(log((p[8]+0.00001)**ew8)+
(Yt1-ew8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[1]))))+
p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5-p[8]+0.00001)))+
(Yt2-ew8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[2]))))+
p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5-p[8]+0.00001)))+
(Yt3-ew8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[3]))))+
p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5-p[8]+0.00001)))+
(Yt4-ew8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[4]))))+
p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5-p[8]+0.00001)))+
(Yt5-ew8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[5]))))+
p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5-p[8]+0.00001)))+
(Yt6-ew8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[6]))))+
p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5-p[8]+0.00001)))+
(Yt7-ew8)*log(((p[5]/(1+exp(-1*(p[6]+p[7]*t[7]))))+
p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5-p[8]+0.00001)))-
(1/p[21])*log(p[21])-lgamma(1/p[21])+
lgamma(Yt1+Yt2+Yt3+Yt4+Yt5+Yt6+Yt7-6*ew8+(1/p[21]))-lgamma(ew8+1)-
lgamma(Yt1-ew8+1)-lgamma(Yt2-ew8+1)-lgamma(Yt3-ew8+1)-
lgamma(Yt4-ew8+1)-lgamma(Yt5-ew8+1)-lgamma(Yt6-ew8+1)-
lgamma(Yt7-ew8+1)-(Yt1+Yt2+Yt3+Yt4+Yt5+Yt6+Yt7-6*ew8+(1/p[21]))*
log((p[5]/(1+exp(-1*(p[6]+p[7]*t[1]))))+
(p[5]/(1+exp(-1*(p[6]+p[7]*t[2]))))+
(p[5]/(1+exp(-1*(p[6]+p[7]*t[3]))))+
(p[5]/(1+exp(-1*(p[6]+p[7]*t[4]))))+
(p[5]/(1+exp(-1*(p[6]+p[7]*t[5]))))+
(p[5]/(1+exp(-1*(p[6]+p[7]*t[6]))))+
(p[5]/(1+exp(-1*(p[6]+p[7]*t[7]))))+
7*(p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5)+0.00001-
6*p[8]+(1/p[21])))*trat2+
(log((p[12]+0.00001)**ew8)+
(Yt1-ew8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[1]))))+
p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5-p[12]+0.00001)))+
(Yt2-ew8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[2]))))+
p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5-p[12]+0.00001)))+
(Yt3-ew8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[3]))))+
p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5-p[12]+0.00001)))+
(Yt4-ew8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[4]))))+
p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5-p[12]+0.00001)))+
(Yt5-ew8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[5]))))+
p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5-p[12]+0.00001)))+
(Yt6-ew8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[6]))))+
p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5-p[12]+0.00001)))+
(Yt7-ew8)*log(((p[9]/(1+exp(-1*(p[10]+p[11]*t[7]))))+
p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5-p[12]+0.00001)))-
(1/p[21])*log(p[21])-lgamma(1/p[21])+

```

```

lgamma(Yt1+Yt2+Yt3+Yt4+Yt5+Yt6+Yt7-6*ew8+(1/p[21]))-lgamma(ew8+1)-
lgamma(Yt1-ew8+1)-lgamma(Yt2-ew8+1)-lgamma(Yt3-ew8+1)-
lgamma(Yt4-ew8+1)-lgamma(Yt5-ew8+1)-lgamma(Yt6-ew8+1)-
lgamma(Yt7-ew8+1)-(Yt1+Yt2+Yt3+Yt4+Yt5+Yt6+Yt7-6*ew8+(1/p[21]))*
log((p[9]/(1+exp(-1*(p[10]+p[11]*t[1])))))+(p[9]/(1+exp(-1*(p[10]+p[11]*t[2]))))+
(p[9]/(1+exp(-1*(p[10]+p[11]*t[3])))))+(p[9]/(1+exp(-1*(p[10]+p[11]*t[4]))))+
(p[9]/(1+exp(-1*(p[10]+p[11]*t[5])))))+(p[9]/(1+exp(-1*(p[10]+p[11]*t[6]))))+
(p[9]/(1+exp(-1*(p[10]+p[11]*t[7]))))+
7*(p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5)+0.00001-
6*p[12]+(1/p[21]))*trat3+
(log((p[16]+0.00001)**ew8)+
(Yt1-ew8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[1]))))+
p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5-p[16]+0.00001)))+
(Yt2-ew8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[2]))))+
p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5-p[16]+0.00001)))+
(Yt3-ew8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[3]))))+
p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5-p[16]+0.00001)))+
(Yt4-ew8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[4]))))+
p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5-p[16]+0.00001)))+
(Yt5-ew8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[5]))))+
p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5-p[16]+0.00001)))+
(Yt6-ew8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[6]))))+
p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5-p[16]+0.00001)))+
(Yt7-ew8)*log(((p[13]/(1+exp(-1*(p[14]+p[15]*t[7]))))+
p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5-p[16]+0.00001)))-
(1/p[21])*log(p[21))-lgamma(1/p[21]))+
lgamma(Yt1+Yt2+Yt3+Yt4+Yt5+Yt6+Yt7-6*ew8+(1/p[21]))-lgamma(ew8+1)-
lgamma(Yt1-ew8+1)-lgamma(Yt2-ew8+1)-lgamma(Yt3-ew8+1)-
lgamma(Yt4-ew8+1)-lgamma(Yt5-ew8+1)-lgamma(Yt6-ew8+1)-lgamma(Yt7-ew8+1)-
(Yt1+Yt2+Yt3+Yt4+Yt5+Yt6+Yt7-6*ew8+(1/p[21]))*
log((p[13]/(1+exp(-1*(p[14]+p[15]*t[1])))))+(p[13]/(1+exp(-1*(p[14]+p[15]*t[2]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[3])))))+(p[13]/(1+exp(-1*(p[14]+p[15]*t[4]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[5])))))+(p[13]/(1+exp(-1*(p[14]+p[15]*t[6]))))+
(p[13]/(1+exp(-1*(p[14]+p[15]*t[7]))))+
7*(p[17]*b2+p[18]*b3+p[19]*b4+p[20]*b5)+0.00001-
6*p[16]+(1/p[21]))*trat4
      lnveroM=sum(lnvero)
      -lnveroM
}

##### Definição do gradiente da função a ser
##### minimizada (-log(verossimilhança))
##### Vetor com -derivada(log(vero)) para cada observação
grd <- deriv(
~((log((p4+0.00001)**ew8)+

```

$$\begin{aligned}
& (Y_{t1}-ew8)*\log(((p1/(1+\exp(-1*(p2+p3*t1))))+ \\
& p17*b2+p18*b3+p19*b4+p20*b5-p4+0.00001)))+ \\
& (Y_{t2}-ew8)*\log(((p1/(1+\exp(-1*(p2+p3*t2))))+ \\
& p17*b2+p18*b3+p19*b4+p20*b5-p4+0.00001)))+ \\
& (Y_{t3}-ew8)*\log(((p1/(1+\exp(-1*(p2+p3*t3))))+ \\
& p17*b2+p18*b3+p19*b4+p20*b5-p4+0.00001)))+ \\
& (Y_{t4}-ew8)*\log(((p1/(1+\exp(-1*(p2+p3*t4))))+ \\
& p17*b2+p18*b3+p19*b4+p20*b5-p4+0.00001)))+ \\
& (Y_{t5}-ew8)*\log(((p1/(1+\exp(-1*(p2+p3*t5))))+ \\
& p17*b2+p18*b3+p19*b4+p20*b5-p4+0.00001)))+ \\
& (Y_{t6}-ew8)*\log(((p1/(1+\exp(-1*(p2+p3*t6))))+ \\
& p17*b2+p18*b3+p19*b4+p20*b5-p4+0.00001)))+ \\
& (Y_{t7}-ew8)*\log(((p1/(1+\exp(-1*(p2+p3*t7))))+ \\
& p17*b2+p18*b3+p19*b4+p20*b5-p4+0.00001)))- \\
& (1/p21)*\log(p21)-\lgamma(1/p21)+ \\
& \lgamma(Y_{t1}+Y_{t2}+Y_{t3}+Y_{t4}+Y_{t5}+Y_{t6}+Y_{t7}-6*ew8+(1/p21))-\lgamma(ew8+1)- \\
& \lgamma(Y_{t1}-ew8+1)-\lgamma(Y_{t2}-ew8+1)-\lgamma(Y_{t3}-ew8+1)- \\
& \lgamma(Y_{t4}-ew8+1)-\lgamma(Y_{t5}-ew8+1)-\lgamma(Y_{t6}-ew8+1)-\lgamma(Y_{t7}-ew8+1)- \\
& (Y_{t1}+Y_{t2}+Y_{t3}+Y_{t4}+Y_{t5}+Y_{t6}+Y_{t7}-6*ew8+(1/p21))* \\
& \log((p1/(1+\exp(-1*(p2+p3*t1))))+(p1/(1+\exp(-1*(p2+p3*t2))))+ \\
& (p1/(1+\exp(-1*(p2+p3*t3))))+(p1/(1+\exp(-1*(p2+p3*t4))))+ \\
& (p1/(1+\exp(-1*(p2+p3*t5))))+(p1/(1+\exp(-1*(p2+p3*t6))))+ \\
& (p1/(1+\exp(-1*(p2+p3*t7))))+ \\
& 7*(p17*b2+p18*b3+p19*b4+p20*b5)+0.00001- \\
& 6*p4+(1/p21))*\text{trat1}+ \\
& (\log((p8+0.00001)**ew8)+ \\
& (Y_{t1}-ew8)*\log(((p5/(1+\exp(-1*(p6+p7*t1))))+ \\
& p17*b2+p18*b3+p19*b4+p20*b5-p8+0.00001)))+ \\
& (Y_{t2}-ew8)*\log(((p5/(1+\exp(-1*(p6+p7*t2))))+ \\
& p17*b2+p18*b3+p19*b4+p20*b5-p8+0.00001)))+ \\
& (Y_{t3}-ew8)*\log(((p5/(1+\exp(-1*(p6+p7*t3))))+ \\
& p17*b2+p18*b3+p19*b4+p20*b5-p8+0.00001)))+ \\
& (Y_{t4}-ew8)*\log(((p5/(1+\exp(-1*(p6+p7*t4))))+ \\
& p17*b2+p18*b3+p19*b4+p20*b5-p8+0.00001)))+ \\
& (Y_{t5}-ew8)*\log(((p5/(1+\exp(-1*(p6+p7*t5))))+ \\
& p17*b2+p18*b3+p19*b4+p20*b5-p8+0.00001)))+ \\
& (Y_{t6}-ew8)*\log(((p5/(1+\exp(-1*(p6+p7*t6))))+ \\
& p17*b2+p18*b3+p19*b4+p20*b5-p8+0.00001)))+ \\
& (Y_{t7}-ew8)*\log(((p5/(1+\exp(-1*(p6+p7*t7))))+ \\
& p17*b2+p18*b3+p19*b4+p20*b5-p8+0.00001)))- \\
& (1/p21)*\log(p21)-\lgamma(1/p21)+ \\
& \lgamma(Y_{t1}+Y_{t2}+Y_{t3}+Y_{t4}+Y_{t5}+Y_{t6}+Y_{t7}-6*ew8+(1/p21))-\lgamma(ew8+1)- \\
& \lgamma(Y_{t1}-ew8+1)-\lgamma(Y_{t2}-ew8+1)-\lgamma(Y_{t3}-ew8+1)- \\
& \lgamma(Y_{t4}-ew8+1)-\lgamma(Y_{t5}-ew8+1)-\lgamma(Y_{t6}-ew8+1)- \\
& \lgamma(Y_{t7}-ew8+1)-(Y_{t1}+Y_{t2}+Y_{t3}+Y_{t4}+Y_{t5}+Y_{t6}+Y_{t7}-6*ew8+(1/p21))*
\end{aligned}$$

$$\begin{aligned}
& \log((p5/(1+\exp(-1*(p6+p7*t1))))+(p5/(1+\exp(-1*(p6+p7*t2))))+ \\
& (p5/(1+\exp(-1*(p6+p7*t3))))+(p5/(1+\exp(-1*(p6+p7*t4))))+ \\
& (p5/(1+\exp(-1*(p6+p7*t5))))+(p5/(1+\exp(-1*(p6+p7*t6))))+ \\
& (p5/(1+\exp(-1*(p6+p7*t7))))+ \\
& 7*(p17*b2+p18*b3+p19*b4+p20*b5)+0.00001- \\
& 6*p8+(1/p21))*\text{trat2}+ \\
& (\log((p12+0.00001)**ew8)+ \\
& (Yt1-ew8)*\log(((p9/(1+\exp(-1*(p10+p11*t1))))+ \\
& p17*b2+p18*b3+p19*b4+p20*b5-p12+0.00001)))+ \\
& (Yt2-ew8)*\log(((p9/(1+\exp(-1*(p10+p11*t2))))+ \\
& p17*b2+p18*b3+p19*b4+p20*b5-p12+0.00001)))+ \\
& (Yt3-ew8)*\log(((p9/(1+\exp(-1*(p10+p11*t3))))+ \\
& p17*b2+p18*b3+p19*b4+p20*b5-p12+0.00001)))+ \\
& (Yt4-ew8)*\log(((p9/(1+\exp(-1*(p10+p11*t4))))+ \\
& p17*b2+p18*b3+p19*b4+p20*b5-p12+0.00001)))+ \\
& (Yt5-ew8)*\log(((p9/(1+\exp(-1*(p10+p11*t5))))+ \\
& p17*b2+p18*b3+p19*b4+p20*b5-p12+0.00001)))+ \\
& (Yt6-ew8)*\log(((p9/(1+\exp(-1*(p10+p11*t6))))+ \\
& p17*b2+p18*b3+p19*b4+p20*b5-p12+0.00001)))+ \\
& (Yt7-ew8)*\log(((p9/(1+\exp(-1*(p10+p11*t7))))+ \\
& p17*b2+p18*b3+p19*b4+p20*b5-p12+0.00001)))- \\
& (1/p21)*\log(p21)-\lgamma(1/p21)+ \\
& \lgamma(Yt1+Yt2+Yt3+Yt4+Yt5+Yt6+Yt7-6*ew8+(1/p21))-\lgamma(ew8+1)- \\
& \lgamma(Yt1-ew8+1)-\lgamma(Yt2-ew8+1)-\lgamma(Yt3-ew8+1)- \\
& \lgamma(Yt4-ew8+1)-\lgamma(Yt5-ew8+1)-\lgamma(Yt6-ew8+1)- \\
& \lgamma(Yt7-ew8+1)-(Yt1+Yt2+Yt3+Yt4+Yt5+Yt6+Yt7-6*ew8+(1/p21))* \\
& \log((p9/(1+\exp(-1*(p10+p11*t1))))+(p9/(1+\exp(-1*(p10+p11*t2))))+ \\
& (p9/(1+\exp(-1*(p10+p11*t3))))+(p9/(1+\exp(-1*(p10+p11*t4))))+ \\
& (p9/(1+\exp(-1*(p10+p11*t5))))+(p9/(1+\exp(-1*(p10+p11*t6))))+ \\
& (p9/(1+\exp(-1*(p10+p11*t7))))+ \\
& 7*(p17*b2+p18*b3+p19*b4+p20*b5)+0.00001- \\
& 6*p12+(1/p21))*\text{trat3}+ \\
& (\log((p16+0.00001)**ew8)+ \\
& (Yt1-ew8)*\log(((p13/(1+\exp(-1*(p14+p15*t1))))+ \\
& p17*b2+p18*b3+p19*b4+p20*b5-p16+0.00001)))+ \\
& (Yt2-ew8)*\log(((p13/(1+\exp(-1*(p14+p15*t2))))+ \\
& p17*b2+p18*b3+p19*b4+p20*b5-p16+0.00001)))+ \\
& (Yt3-ew8)*\log(((p13/(1+\exp(-1*(p14+p15*t3))))+ \\
& p17*b2+p18*b3+p19*b4+p20*b5-p16+0.00001)))+ \\
& (Yt4-ew8)*\log(((p13/(1+\exp(-1*(p14+p15*t4))))+ \\
& p17*b2+p18*b3+p19*b4+p20*b5-p16+0.00001)))+ \\
& (Yt5-ew8)*\log(((p13/(1+\exp(-1*(p14+p15*t5))))+ \\
& p17*b2+p18*b3+p19*b4+p20*b5-p16+0.00001)))+ \\
& (Yt6-ew8)*\log(((p13/(1+\exp(-1*(p14+p15*t6))))+ \\
& p17*b2+p18*b3+p19*b4+p20*b5-p16+0.00001)))+
\end{aligned}$$

```

(Yt7-ew8)*log((((p13/(1+exp(-1*(p14+p15*t7)))))+
p17*b2+p18*b3+p19*b4+p20*b5-p16+0.00001)))-
(1/p21)*log(p21)-lgamma(1/p21)+
lgamma(Yt1+Yt2+Yt3+Yt4+Yt5+Yt6+Yt7-6*ew8+(1/p21))-lgamma(ew8+1)-
lgamma(Yt1-ew8+1)-lgamma(Yt2-ew8+1)-lgamma(Yt3-ew8+1)-
lgamma(Yt4-ew8+1)-lgamma(Yt5-ew8+1)-lgamma(Yt6-ew8+1)-
lgamma(Yt7-ew8+1)-(Yt1+Yt2+Yt3+Yt4+Yt5+Yt6+Yt7-6*ew8+(1/p21))*
log((p13/(1+exp(-1*(p14+p15*t1))))+(p13/(1+exp(-1*(p14+p15*t2))))+
(p13/(1+exp(-1*(p14+p15*t3))))+(p13/(1+exp(-1*(p14+p15*t4))))+
(p13/(1+exp(-1*(p14+p15*t5))))+(p13/(1+exp(-1*(p14+p15*t6))))+
(p13/(1+exp(-1*(p14+p15*t7))))+
7*(p17*b2+p18*b3+p19*b4+p20*b5)+0.00001-6*p16+(1/p21))*trat4),
c("p1","p2","p3","p4","p5","p6","p7","p8","p9","p10","p11","p12",
"p13","p14","p15","p16","p17","p18","p19","p20","p21"),
function(p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11,p12,p13,p14,
p15,p16,p17,p18,p19,p20,p21,t1,t2,t3,t4,t5,t6,t7) NULL)

sgrd <- function(p,Yt1,Yt2,Yt3,Yt4,Yt5,Yt6,Yt7,ew8) {
  p1=p[1]; p2=p[2]; p3=p[3]; p4=p[4]; p5=p[5]; p6=p[6]; p7=p[7]
  p8=p[8]; p9=p[9]; p10=p[10]; p11=p[11]; p12=p[12]; p13=p[13]
  p14=p[14]; p15=p[15]; p16=p[16]; p17=p[17]; p18=p[18]
  p19=p[19]; p20=p[20]; p21=p[21]
  t1=t[1]; t2=t[2]; t3=t[3]; t4=t[4]; t5=t[5]; t6=t[6]; t7=t[7]
  colSums(attr(grd(p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11,p12,
                    p13,p14,p15,p16,p17,p18,p19,p20,p21,
                    t1,t2,t3,t4,t5,t6,t7),"gradient"))
}

##### Início do Processo de Simulação #####
W11=matrix(0,nrow=n,ncol=1); W12=matrix(0,nrow=n,ncol=1)
W13=matrix(0,nrow=n,ncol=1); W14=matrix(0,nrow=n,ncol=1)
W15=matrix(0,nrow=n,ncol=1); W16=matrix(0,nrow=n,ncol=1)
W17=matrix(0,nrow=n,ncol=1); W18=matrix(0,nrow=n,ncol=1)

W21=matrix(0,nrow=n,ncol=1); W22=matrix(0,nrow=n,ncol=1)
W23=matrix(0,nrow=n,ncol=1); W24=matrix(0,nrow=n,ncol=1)
W25=matrix(0,nrow=n,ncol=1); W26=matrix(0,nrow=n,ncol=1)
W27=matrix(0,nrow=n,ncol=1); W28=matrix(0,nrow=n,ncol=1)

W31=matrix(0,nrow=n,ncol=1); W32=matrix(0,nrow=n,ncol=1)
W33=matrix(0,nrow=n,ncol=1); W34=matrix(0,nrow=n,ncol=1)
W35=matrix(0,nrow=n,ncol=1); W36=matrix(0,nrow=n,ncol=1)
W37=matrix(0,nrow=n,ncol=1); W38=matrix(0,nrow=n,ncol=1)

W41=matrix(0,nrow=n,ncol=1); W42=matrix(0,nrow=n,ncol=1)

```

```

W43=matrix(0,nrow=n,ncol=1); W44=matrix(0,nrow=n,ncol=1)
W45=matrix(0,nrow=n,ncol=1); W46=matrix(0,nrow=n,ncol=1)
W47=matrix(0,nrow=n,ncol=1); W48=matrix(0,nrow=n,ncol=1)

set.seed(33); smt=runif(1000)
smt=round(smt,digits=5); smt=smt*100000
result=matrix(0,nrow=2000,ncol=29)
k=1; nsimul=1
system.time(
while (nsimul <= 1000){
set.seed(smt[nsimul])

##### Gerando os efeitos aleatórios (Zij)
##### seguindo distribuição Gama(1/psi,1/psi)
# Efeito aleatório para o tratamento 1
Z1=rgamma(n, shape=1/psip, scale=psip)
Z1b=rep(Z1,7)
# Efeito aleatório para o tratamento 2
Z2=rgamma(n, shape=1/psip, scale=psip)
Z2b=rep(Z1,7)
# Efeito aleatório para o tratamento 3
Z3=rgamma(n, shape=1/psip, scale=psip)
Z3b=rep(Z1,7)
# Efeito aleatório para o tratamento 4
Z4=rgamma(n, shape=1/psip, scale=psip)
Z4b=rep(Z1,7)

Z=c(Z1,Z2,Z3,Z4)
Zcomp=c(Z1b,Z2b,Z3b,Z4b)

##### Gerando os valores das variáveis latentes
##### condicionais aos valores de Zij
for (i in 1:n) {
#Variáveis latentes para o tratamento 1
W11[i]=rpois(1,(fmu1[1]+betap[i]-theta18p)*Z1[i])
W12[i]=rpois(1,(fmu1[2]+betap[i]-theta18p)*Z1[i])
W13[i]=rpois(1,(fmu1[3]+betap[i]-theta18p)*Z1[i])
W14[i]=rpois(1,(fmu1[4]+betap[i]-theta18p)*Z1[i])
W15[i]=rpois(1,(fmu1[5]+betap[i]-theta18p)*Z1[i])
W16[i]=rpois(1,(fmu1[6]+betap[i]-theta18p)*Z1[i])
W17[i]=rpois(1,(fmu1[7]+betap[i]-theta18p)*Z1[i])
W18[i]=rpois(1,theta18p*Z1[i])

#Variáveis latentes para o tratamento 2
W21[i]=rpois(1,(fmu2[1]+betap[i]-theta28p)*Z2[i])

```

```

W22[i]=rpois(1,(fmu2[2]+betap[i]-theta28p)*Z2[i])
W23[i]=rpois(1,(fmu2[3]+betap[i]-theta28p)*Z2[i])
W24[i]=rpois(1,(fmu2[4]+betap[i]-theta28p)*Z2[i])
W25[i]=rpois(1,(fmu2[5]+betap[i]-theta28p)*Z2[i])
W26[i]=rpois(1,(fmu2[6]+betap[i]-theta28p)*Z2[i])
W27[i]=rpois(1,(fmu2[7]+betap[i]-theta28p)*Z2[i])
W28[i]=rpois(1,theta28p*Z2[i])

```

```

#Variáveis latentes para o tratamento 3

```

```

W31[i]=rpois(1,(fmu3[1]+betap[i]-theta38p)*Z3[i])
W32[i]=rpois(1,(fmu3[2]+betap[i]-theta38p)*Z3[i])
W33[i]=rpois(1,(fmu3[3]+betap[i]-theta38p)*Z3[i])
W34[i]=rpois(1,(fmu3[4]+betap[i]-theta38p)*Z3[i])
W35[i]=rpois(1,(fmu3[5]+betap[i]-theta38p)*Z3[i])
W36[i]=rpois(1,(fmu3[6]+betap[i]-theta38p)*Z3[i])
W37[i]=rpois(1,(fmu3[7]+betap[i]-theta38p)*Z3[i])
W38[i]=rpois(1,theta38p*Z3[i])

```

```

#Variáveis latentes para o tratamento 4

```

```

W41[i]=rpois(1,(fmu4[1]+betap[i]-theta48p)*Z4[i])
W42[i]=rpois(1,(fmu4[2]+betap[i]-theta48p)*Z4[i])
W43[i]=rpois(1,(fmu4[3]+betap[i]-theta48p)*Z4[i])
W44[i]=rpois(1,(fmu4[4]+betap[i]-theta48p)*Z4[i])
W45[i]=rpois(1,(fmu4[5]+betap[i]-theta48p)*Z4[i])
W46[i]=rpois(1,(fmu4[6]+betap[i]-theta48p)*Z4[i])
W47[i]=rpois(1,(fmu4[7]+betap[i]-theta48p)*Z4[i])
W48[i]=rpois(1,theta48p*Z4[i])
}

```

```

#### Gerando os valores observados

```

```

##### condicionais aos valores de Zij

```

```

#Variáveis observadas para o tratamento 1

```

```

Y11=W11+W18; Y12=W12+W18; Y13=W13+W18; Y14=W14+W18
Y15=W15+W18; Y16=W16+W18; Y17=W17+W18

```

```

Y1=cbind(Y11,Y12,Y13,Y14,Y15,Y16,Y17)
Ytrat1=rbind(t(t(Y11)),t(t(Y12)),t(t(Y13)),t(t(Y14)),t(t(Y15)),
             t(t(Y16)),t(t(Y17)))

```

```

#Variáveis observadas para o tratamento 2

```

```

Y21=W21+W28; Y22=W22+W28; Y23=W23+W28; Y24=W24+W28
Y25=W25+W28; Y26=W26+W28; Y27=W27+W28

```

```

Y2=cbind(Y21,Y22,Y23,Y24,Y25,Y26,Y27)

```



```

Ytrat2=rbind(t(t(Y21)),t(t(Y22)),t(t(Y23)),t(t(Y24)),t(t(Y25)),
             t(t(Y26)),t(t(Y27)))

#Variáveis observadas para o tratamento 3
Y31=W31+W38; Y32=W32+W38; Y33=W33+W38; Y34=W34+W38
Y35=W35+W38; Y36=W36+W38; Y37=W37+W38

Y3=cbind(Y31,Y32,Y33,Y34,Y35,Y36,Y37)
Ytrat3=rbind(t(t(Y31)),t(t(Y32)),t(t(Y33)),t(t(Y34)),t(t(Y35)),
             t(t(Y36)),t(t(Y37)))

#Variáveis observadas para o tratamento 4
Y41=W41+W48; Y42=W42+W48; Y43=W43+W48; Y44=W44+W48
Y45=W45+W48; Y46=W46+W48; Y47=W47+W48

Y4=cbind(Y41,Y42,Y43,Y44,Y45,Y46,Y47)
Ytrat4=rbind(t(t(Y41)),t(t(Y42)),t(t(Y43)),t(t(Y44)),t(t(Y45)),
             t(t(Y46)),t(t(Y47)))

Y=rbind(Y1,Y2,Y3,Y4)
Yt1=Y[,1]; Yt2=Y[,2]; Yt3=Y[,3]; Yt4=Y[,4]; Yt5=Y[,5]; Yt6=Y[,6]; Yt7=Y[,7]

tempo=c(rep(t[1],n),rep(t[2],n),rep(t[3],n),rep(t[4],n),rep(t[5],n),
        rep(t[6],n),rep(t[7],n))
Ytrat=data.frame(tempo,Ytrat1,Ytrat2,Ytrat3,Ytrat4)

#Encontrando valores iniciais para os parâmetros da regressão
init1 <- nlreg( Ytrat1 ~ (alfa1/(1+exp(-1*(delta1+gama1*tempo)))),
              start = c(alfa1=200, delta1=-1, gama1=0.05), data = Ytrat)
init2 <- nlreg( Ytrat2 ~ (alfa2/(1+exp(-1*(delta2+gama2*tempo)))),
              start = c(alfa2=200, delta2=-1, gama2=0.05), data = Ytrat)
init3 <- nlreg( Ytrat3 ~ (alfa3/(1+exp(-1*(delta3+gama3*tempo)))),
              start = c(alfa3=200, delta3=-1, gama3=0.05), data = Ytrat)
init4 <- nlreg( Ytrat4 ~ (alfa4/(1+exp(-1*(delta4+gama4*tempo)))),
              start = c(alfa4=200, delta4=-1, gama4=0.05), data = Ytrat)

alfa1=init1$coef[1]; delta1=init1$coef[2]; gama1=init1$coef[3]
alfa1_ini=alfa1; delta1_ini=delta1; gama1_ini=gama1
alfa2=init2$coef[1]; delta2=init2$coef[2]; gama2=init2$coef[3]
alfa2_ini=alfa2; delta2_ini=delta2; gama2_ini=gama2
alfa3=init3$coef[1]; delta3=init3$coef[2]; gama3=init3$coef[3]
alfa3_ini=alfa3; delta3_ini=delta3; gama3_ini=gama3
alfa4=init4$coef[1]; delta4=init4$coef[2]; gama4=init4$coef[3]
alfa4_ini=alfa4; delta4_ini=delta4; gama4_ini=gama4

```



```
# Procedimento para estimar o efeito de bloco para usar como chute inicial
```

```
Yt1.bloco=cbind(trat,bloco,Yt1); Yt2.bloco=cbind(trat,bloco,Yt2)
Yt3.bloco=cbind(trat,bloco,Yt3); Yt4.bloco=cbind(trat,bloco,Yt4)
Yt5.bloco=cbind(trat,bloco,Yt5); Yt6.bloco=cbind(trat,bloco,Yt6)
Yt7.bloco=cbind(trat,bloco,Yt7)
```

```
dados.bloco=data.frame(rbind(Yt1.bloco,Yt2.bloco,Yt3.bloco,
                             Yt4.bloco,Yt5.bloco,Yt6.bloco,Yt7.bloco))
```

```
tp=c(rep(1,20),rep(2,20),rep(3,20),rep(4,20),rep(5,20),rep(6,20),rep(7,20))
```

```
dados.bloco$tp=tp
init.bloco = glm(Yt1 ~ factor(trat)+factor(bloco),data=dados.bloco)
b=c(0,init.bloco$coef[5:8])
```

```
beta1=b[1]
beta1_ini=beta1
```

```
beta2=max(b[2],-min(alfa1/(1+exp(-1*(delta1+gama1*t[1])))-1,
                    alfa1/(1+exp(-1*(delta1+gama1*t[2])))-1,
                    alfa1/(1+exp(-1*(delta1+gama1*t[3])))-1,
                    alfa1/(1+exp(-1*(delta1+gama1*t[4])))-1,
                    alfa1/(1+exp(-1*(delta1+gama1*t[5])))-1,
                    alfa1/(1+exp(-1*(delta1+gama1*t[6])))-1,
                    alfa1/(1+exp(-1*(delta1+gama1*t[7])))-1,
                    alfa2/(1+exp(-1*(delta2+gama2*t[1])))-1,
                    alfa2/(1+exp(-1*(delta2+gama2*t[2])))-1,
                    alfa2/(1+exp(-1*(delta2+gama2*t[3])))-1,
                    alfa2/(1+exp(-1*(delta2+gama2*t[4])))-1,
                    alfa2/(1+exp(-1*(delta2+gama2*t[5])))-1,
                    alfa2/(1+exp(-1*(delta2+gama2*t[6])))-1,
                    alfa2/(1+exp(-1*(delta2+gama2*t[7])))-1,
                    alfa3/(1+exp(-1*(delta3+gama3*t[1])))-1,
                    alfa3/(1+exp(-1*(delta3+gama3*t[2])))-1,
                    alfa3/(1+exp(-1*(delta3+gama3*t[3])))-1,
                    alfa3/(1+exp(-1*(delta3+gama3*t[4])))-1,
                    alfa3/(1+exp(-1*(delta3+gama3*t[5])))-1,
                    alfa3/(1+exp(-1*(delta3+gama3*t[6])))-1,
                    alfa3/(1+exp(-1*(delta3+gama3*t[7])))-1,
                    alfa4/(1+exp(-1*(delta4+gama4*t[1])))-1,
                    alfa4/(1+exp(-1*(delta4+gama4*t[2])))-1,
                    alfa4/(1+exp(-1*(delta4+gama4*t[3])))-1,
                    alfa4/(1+exp(-1*(delta4+gama4*t[4])))-1,
                    alfa4/(1+exp(-1*(delta4+gama4*t[5])))-1,
```

```

        alfa4/(1+exp(-1*(delta4+gama4*t[6]))) -1,
        alfa4/(1+exp(-1*(delta4+gama4*t[7]))) -1))
beta2_ini=beta2

beta3=max(b[3],-min(alfa1/(1+exp(-1*(delta1+gama1*t[1]))) -1,
        alfa1/(1+exp(-1*(delta1+gama1*t[2]))) -1,
        alfa1/(1+exp(-1*(delta1+gama1*t[3]))) -1,
        alfa1/(1+exp(-1*(delta1+gama1*t[4]))) -1,
        alfa1/(1+exp(-1*(delta1+gama1*t[5]))) -1,
        alfa1/(1+exp(-1*(delta1+gama1*t[6]))) -1,
        alfa1/(1+exp(-1*(delta1+gama1*t[7]))) -1,
        alfa2/(1+exp(-1*(delta2+gama2*t[1]))) -1,
        alfa2/(1+exp(-1*(delta2+gama2*t[2]))) -1,
        alfa2/(1+exp(-1*(delta2+gama2*t[3]))) -1,
        alfa2/(1+exp(-1*(delta2+gama2*t[4]))) -1,
        alfa2/(1+exp(-1*(delta2+gama2*t[5]))) -1,
        alfa2/(1+exp(-1*(delta2+gama2*t[6]))) -1,
        alfa2/(1+exp(-1*(delta2+gama2*t[7]))) -1,
        alfa3/(1+exp(-1*(delta3+gama3*t[1]))) -1,
        alfa3/(1+exp(-1*(delta3+gama3*t[2]))) -1,
        alfa3/(1+exp(-1*(delta3+gama3*t[3]))) -1,
        alfa3/(1+exp(-1*(delta3+gama3*t[4]))) -1,
        alfa3/(1+exp(-1*(delta3+gama3*t[5]))) -1,
        alfa3/(1+exp(-1*(delta3+gama3*t[6]))) -1,
        alfa3/(1+exp(-1*(delta3+gama3*t[7]))) -1,
        alfa4/(1+exp(-1*(delta4+gama4*t[1]))) -1,
        alfa4/(1+exp(-1*(delta4+gama4*t[2]))) -1,
        alfa4/(1+exp(-1*(delta4+gama4*t[3]))) -1,
        alfa4/(1+exp(-1*(delta4+gama4*t[4]))) -1,
        alfa4/(1+exp(-1*(delta4+gama4*t[5]))) -1,
        alfa4/(1+exp(-1*(delta4+gama4*t[6]))) -1,
        alfa4/(1+exp(-1*(delta4+gama4*t[7]))) -1))
beta3_ini=beta3

beta4=max(b[4],-min(alfa1/(1+exp(-1*(delta1+gama1*t[1]))) -1,
        alfa1/(1+exp(-1*(delta1+gama1*t[2]))) -1,
        alfa1/(1+exp(-1*(delta1+gama1*t[3]))) -1,
        alfa1/(1+exp(-1*(delta1+gama1*t[4]))) -1,
        alfa1/(1+exp(-1*(delta1+gama1*t[5]))) -1,
        alfa1/(1+exp(-1*(delta1+gama1*t[6]))) -1,
        alfa1/(1+exp(-1*(delta1+gama1*t[7]))) -1,
        alfa2/(1+exp(-1*(delta2+gama2*t[1]))) -1,
        alfa2/(1+exp(-1*(delta2+gama2*t[2]))) -1,
        alfa2/(1+exp(-1*(delta2+gama2*t[3]))) -1,
        alfa2/(1+exp(-1*(delta2+gama2*t[4]))) -1,

```

```

alfa2/(1+exp(-1*(delta2+gama2*t[5]))) -1,
alfa2/(1+exp(-1*(delta2+gama2*t[6]))) -1,
alfa2/(1+exp(-1*(delta2+gama2*t[7]))) -1,
alfa3/(1+exp(-1*(delta3+gama3*t[1]))) -1,
alfa3/(1+exp(-1*(delta3+gama3*t[2]))) -1,
alfa3/(1+exp(-1*(delta3+gama3*t[3]))) -1,
alfa3/(1+exp(-1*(delta3+gama3*t[4]))) -1,
alfa3/(1+exp(-1*(delta3+gama3*t[5]))) -1,
alfa3/(1+exp(-1*(delta3+gama3*t[6]))) -1,
alfa3/(1+exp(-1*(delta3+gama3*t[7]))) -1,
alfa4/(1+exp(-1*(delta4+gama4*t[1]))) -1,
alfa4/(1+exp(-1*(delta4+gama4*t[2]))) -1,
alfa4/(1+exp(-1*(delta4+gama4*t[3]))) -1,
alfa4/(1+exp(-1*(delta4+gama4*t[4]))) -1,
alfa4/(1+exp(-1*(delta4+gama4*t[5]))) -1,
alfa4/(1+exp(-1*(delta4+gama4*t[6]))) -1,
alfa4/(1+exp(-1*(delta4+gama4*t[7]))) -1,
beta4_ini=beta4

beta5=max(b[5], -min(alfa1/(1+exp(-1*(delta1+gama1*t[1]))) -1,
    alfa1/(1+exp(-1*(delta1+gama1*t[2]))) -1,
    alfa1/(1+exp(-1*(delta1+gama1*t[3]))) -1,
    alfa1/(1+exp(-1*(delta1+gama1*t[4]))) -1,
    alfa1/(1+exp(-1*(delta1+gama1*t[5]))) -1,
    alfa1/(1+exp(-1*(delta1+gama1*t[6]))) -1,
    alfa1/(1+exp(-1*(delta1+gama1*t[7]))) -1,
    alfa2/(1+exp(-1*(delta2+gama2*t[1]))) -1,
    alfa2/(1+exp(-1*(delta2+gama2*t[2]))) -1,
    alfa2/(1+exp(-1*(delta2+gama2*t[3]))) -1,
    alfa2/(1+exp(-1*(delta2+gama2*t[4]))) -1,
    alfa2/(1+exp(-1*(delta2+gama2*t[5]))) -1,
    alfa2/(1+exp(-1*(delta2+gama2*t[6]))) -1,
    alfa2/(1+exp(-1*(delta2+gama2*t[7]))) -1,
    alfa3/(1+exp(-1*(delta3+gama3*t[1]))) -1,
    alfa3/(1+exp(-1*(delta3+gama3*t[2]))) -1,
    alfa3/(1+exp(-1*(delta3+gama3*t[3]))) -1,
    alfa3/(1+exp(-1*(delta3+gama3*t[4]))) -1,
    alfa3/(1+exp(-1*(delta3+gama3*t[5]))) -1,
    alfa3/(1+exp(-1*(delta3+gama3*t[6]))) -1,
    alfa3/(1+exp(-1*(delta3+gama3*t[7]))) -1,
    alfa4/(1+exp(-1*(delta4+gama4*t[1]))) -1,
    alfa4/(1+exp(-1*(delta4+gama4*t[2]))) -1,
    alfa4/(1+exp(-1*(delta4+gama4*t[3]))) -1,
    alfa4/(1+exp(-1*(delta4+gama4*t[4]))) -1,
    alfa4/(1+exp(-1*(delta4+gama4*t[5]))) -1,

```

```

    alfa4/(1+exp(-1*(delta4+gama4*t[6])))-1,
    alfa4/(1+exp(-1*(delta4+gama4*t[7]))))
beta5_ini=beta5

#Encontrando os valores iniciais para os parâmetros de covariância
if (cov(Y11,Y12)>0) cov12_1=cov(Y11,Y12) else cov12_1=0
if (cov(Y11,Y13)>0) cov13_1=cov(Y11,Y13) else cov13_1=0
if (cov(Y11,Y14)>0) cov14_1=cov(Y11,Y14) else cov14_1=0
if (cov(Y11,Y15)>0) cov15_1=cov(Y11,Y15) else cov15_1=0
if (cov(Y11,Y16)>0) cov16_1=cov(Y11,Y16) else cov16_1=0
if (cov(Y11,Y17)>0) cov17_1=cov(Y11,Y17) else cov17_1=0
if (cov(Y12,Y13)>0) cov23_1=cov(Y12,Y13) else cov23_1=0
if (cov(Y12,Y14)>0) cov24_1=cov(Y12,Y14) else cov24_1=0
if (cov(Y12,Y15)>0) cov25_1=cov(Y12,Y15) else cov25_1=0
if (cov(Y12,Y16)>0) cov26_1=cov(Y12,Y16) else cov26_1=0
if (cov(Y12,Y17)>0) cov27_1=cov(Y12,Y17) else cov27_1=0
if (cov(Y13,Y14)>0) cov34_1=cov(Y13,Y14) else cov34_1=0
if (cov(Y13,Y15)>0) cov35_1=cov(Y13,Y15) else cov35_1=0
if (cov(Y13,Y16)>0) cov36_1=cov(Y13,Y16) else cov36_1=0
if (cov(Y13,Y17)>0) cov37_1=cov(Y13,Y17) else cov37_1=0
if (cov(Y14,Y15)>0) cov45_1=cov(Y14,Y15) else cov45_1=0
if (cov(Y14,Y16)>0) cov46_1=cov(Y14,Y16) else cov46_1=0
if (cov(Y14,Y17)>0) cov47_1=cov(Y14,Y17) else cov47_1=0
if (cov(Y15,Y16)>0) cov56_1=cov(Y15,Y16) else cov56_1=0
if (cov(Y15,Y17)>0) cov57_1=cov(Y15,Y17) else cov57_1=0
if (cov(Y16,Y17)>0) cov67_1=cov(Y16,Y17) else cov67_1=0

cov1=c(cov12_1,cov13_1,cov14_1,cov15_1,cov16_1,cov17_1,cov23_1,cov24_1,
      cov25_1,cov26_1,cov27_1,cov34_1,cov35_1,cov36_1,cov37_1,cov45_1,
      cov46_1,cov47_1,cov56_1,cov57_1,cov67_1)

theta18=min(mean(cov1),(alfa1/(1+exp(-1*(delta1+gama1*t[1]))))+beta1,
            (alfa1/(1+exp(-1*(delta1+gama1*t[1]))))+beta2,
            (alfa1/(1+exp(-1*(delta1+gama1*t[1]))))+beta3,
            (alfa1/(1+exp(-1*(delta1+gama1*t[1]))))+beta4,
            (alfa1/(1+exp(-1*(delta1+gama1*t[1]))))+beta5,
            (alfa1/(1+exp(-1*(delta1+gama1*t[2]))))+beta1,
            (alfa1/(1+exp(-1*(delta1+gama1*t[2]))))+beta2,
            (alfa1/(1+exp(-1*(delta1+gama1*t[2]))))+beta3,
            (alfa1/(1+exp(-1*(delta1+gama1*t[2]))))+beta4,
            (alfa1/(1+exp(-1*(delta1+gama1*t[2]))))+beta5,
            (alfa1/(1+exp(-1*(delta1+gama1*t[3]))))+beta1,
            (alfa1/(1+exp(-1*(delta1+gama1*t[3]))))+beta2,
            (alfa1/(1+exp(-1*(delta1+gama1*t[3]))))+beta3,
            (alfa1/(1+exp(-1*(delta1+gama1*t[3]))))+beta4,

```

```

(alfa1/(1+exp(-1*(delta1+gama1*t[3]))))+beta5,
(alfa1/(1+exp(-1*(delta1+gama1*t[4]))))+beta1,
(alfa1/(1+exp(-1*(delta1+gama1*t[4]))))+beta2,
(alfa1/(1+exp(-1*(delta1+gama1*t[4]))))+beta3,
(alfa1/(1+exp(-1*(delta1+gama1*t[4]))))+beta4,
(alfa1/(1+exp(-1*(delta1+gama1*t[4]))))+beta5,
(alfa1/(1+exp(-1*(delta1+gama1*t[5]))))+beta1,
(alfa1/(1+exp(-1*(delta1+gama1*t[5]))))+beta2,
(alfa1/(1+exp(-1*(delta1+gama1*t[5]))))+beta3,
(alfa1/(1+exp(-1*(delta1+gama1*t[5]))))+beta4,
(alfa1/(1+exp(-1*(delta1+gama1*t[5]))))+beta5,
(alfa1/(1+exp(-1*(delta1+gama1*t[6]))))+beta1,
(alfa1/(1+exp(-1*(delta1+gama1*t[6]))))+beta2,
(alfa1/(1+exp(-1*(delta1+gama1*t[6]))))+beta3,
(alfa1/(1+exp(-1*(delta1+gama1*t[6]))))+beta4,
(alfa1/(1+exp(-1*(delta1+gama1*t[6]))))+beta5,
(alfa1/(1+exp(-1*(delta1+gama1*t[7]))))+beta1,
(alfa1/(1+exp(-1*(delta1+gama1*t[7]))))+beta2,
(alfa1/(1+exp(-1*(delta1+gama1*t[7]))))+beta3,
(alfa1/(1+exp(-1*(delta1+gama1*t[7]))))+beta4,
(alfa1/(1+exp(-1*(delta1+gama1*t[7]))))+beta5)
theta18_ini=theta18

if (cov(Y21,Y22)>0) cov12_2=cov(Y21,Y22) else cov12_2=0
if (cov(Y21,Y23)>0) cov13_2=cov(Y21,Y23) else cov13_2=0
if (cov(Y21,Y24)>0) cov14_2=cov(Y21,Y24) else cov14_2=0
if (cov(Y21,Y25)>0) cov15_2=cov(Y21,Y25) else cov15_2=0
if (cov(Y21,Y26)>0) cov16_2=cov(Y21,Y26) else cov16_2=0
if (cov(Y21,Y27)>0) cov17_2=cov(Y21,Y27) else cov17_2=0
if (cov(Y22,Y23)>0) cov23_2=cov(Y22,Y23) else cov23_2=0
if (cov(Y22,Y24)>0) cov24_2=cov(Y22,Y24) else cov24_2=0
if (cov(Y22,Y25)>0) cov25_2=cov(Y22,Y25) else cov25_2=0
if (cov(Y22,Y26)>0) cov26_2=cov(Y22,Y26) else cov26_2=0
if (cov(Y22,Y27)>0) cov27_2=cov(Y22,Y27) else cov27_2=0
if (cov(Y23,Y24)>0) cov34_2=cov(Y23,Y24) else cov34_2=0
if (cov(Y23,Y25)>0) cov35_2=cov(Y23,Y25) else cov35_2=0
if (cov(Y23,Y26)>0) cov36_2=cov(Y23,Y26) else cov36_2=0
if (cov(Y23,Y27)>0) cov37_2=cov(Y23,Y27) else cov37_2=0
if (cov(Y24,Y25)>0) cov45_2=cov(Y24,Y25) else cov45_2=0
if (cov(Y24,Y26)>0) cov46_2=cov(Y24,Y26) else cov46_2=0
if (cov(Y24,Y27)>0) cov47_2=cov(Y24,Y27) else cov47_2=0
if (cov(Y25,Y26)>0) cov56_2=cov(Y25,Y26) else cov56_2=0
if (cov(Y25,Y27)>0) cov57_2=cov(Y25,Y27) else cov57_2=0
if (cov(Y26,Y27)>0) cov67_2=cov(Y26,Y27) else cov67_2=0

```

```
cov2=c(cov12_2,cov13_2,cov14_2,cov15_2,cov16_2,cov17_2,cov23_2,cov24_2,
        cov25_2,cov26_2,cov27_2,cov34_2,cov35_2,cov36_2,cov37_2,cov45_2,
        cov46_2,cov47_2,cov56_2,cov57_2,cov67_2)
```

```
theta28=min(mean(cov2),(alfa2/(1+exp(-1*(delta2+gama2*t[1]))))+beta1,
            (alfa2/(1+exp(-1*(delta2+gama2*t[1]))))+beta2,
            (alfa2/(1+exp(-1*(delta2+gama2*t[1]))))+beta3,
            (alfa2/(1+exp(-1*(delta2+gama2*t[1]))))+beta4,
            (alfa2/(1+exp(-1*(delta2+gama2*t[1]))))+beta5,
            (alfa2/(1+exp(-1*(delta2+gama2*t[2]))))+beta1,
            (alfa2/(1+exp(-1*(delta2+gama2*t[2]))))+beta2,
            (alfa2/(1+exp(-1*(delta2+gama2*t[2]))))+beta3,
            (alfa2/(1+exp(-1*(delta2+gama2*t[2]))))+beta4,
            (alfa2/(1+exp(-1*(delta2+gama2*t[2]))))+beta5,
            (alfa2/(1+exp(-1*(delta2+gama2*t[3]))))+beta1,
            (alfa2/(1+exp(-1*(delta2+gama2*t[3]))))+beta2,
            (alfa2/(1+exp(-1*(delta2+gama2*t[3]))))+beta3,
            (alfa2/(1+exp(-1*(delta2+gama2*t[3]))))+beta4,
            (alfa2/(1+exp(-1*(delta2+gama2*t[3]))))+beta5,
            (alfa2/(1+exp(-1*(delta2+gama2*t[4]))))+beta1,
            (alfa2/(1+exp(-1*(delta2+gama2*t[4]))))+beta2,
            (alfa2/(1+exp(-1*(delta2+gama2*t[4]))))+beta3,
            (alfa2/(1+exp(-1*(delta2+gama2*t[4]))))+beta4,
            (alfa2/(1+exp(-1*(delta2+gama2*t[4]))))+beta5,
            (alfa2/(1+exp(-1*(delta2+gama2*t[5]))))+beta1,
            (alfa2/(1+exp(-1*(delta2+gama2*t[5]))))+beta2,
            (alfa2/(1+exp(-1*(delta2+gama2*t[5]))))+beta3,
            (alfa2/(1+exp(-1*(delta2+gama2*t[5]))))+beta4,
            (alfa2/(1+exp(-1*(delta2+gama2*t[5]))))+beta5,
            (alfa2/(1+exp(-1*(delta2+gama2*t[6]))))+beta1,
            (alfa2/(1+exp(-1*(delta2+gama2*t[6]))))+beta2,
            (alfa2/(1+exp(-1*(delta2+gama2*t[6]))))+beta3,
            (alfa2/(1+exp(-1*(delta2+gama2*t[6]))))+beta4,
            (alfa2/(1+exp(-1*(delta2+gama2*t[6]))))+beta5,
            (alfa2/(1+exp(-1*(delta2+gama2*t[7]))))+beta1,
            (alfa2/(1+exp(-1*(delta2+gama2*t[7]))))+beta2,
            (alfa2/(1+exp(-1*(delta2+gama2*t[7]))))+beta3,
            (alfa2/(1+exp(-1*(delta2+gama2*t[7]))))+beta4,
            (alfa2/(1+exp(-1*(delta2+gama2*t[7]))))+beta5)
theta28_ini=theta28
```

```
if (cov(Y31,Y32)>0) cov12_3=cov(Y31,Y32) else cov12_3=0
if (cov(Y31,Y33)>0) cov13_3=cov(Y31,Y33) else cov13_3=0
if (cov(Y31,Y34)>0) cov14_3=cov(Y31,Y34) else cov14_3=0
if (cov(Y31,Y35)>0) cov15_3=cov(Y31,Y35) else cov15_3=0
```

```

if (cov(Y31,Y36)>0) cov16_3=cov(Y31,Y36) else cov16_3=0
if (cov(Y31,Y37)>0) cov17_3=cov(Y31,Y37) else cov17_3=0
if (cov(Y32,Y33)>0) cov23_3=cov(Y32,Y33) else cov23_3=0
if (cov(Y32,Y34)>0) cov24_3=cov(Y32,Y34) else cov24_3=0
if (cov(Y32,Y35)>0) cov25_3=cov(Y32,Y35) else cov25_3=0
if (cov(Y32,Y36)>0) cov26_3=cov(Y32,Y36) else cov26_3=0
if (cov(Y32,Y37)>0) cov27_3=cov(Y32,Y37) else cov27_3=0
if (cov(Y33,Y34)>0) cov34_3=cov(Y33,Y34) else cov34_3=0
if (cov(Y33,Y35)>0) cov35_3=cov(Y33,Y35) else cov35_3=0
if (cov(Y33,Y36)>0) cov36_3=cov(Y33,Y36) else cov36_3=0
if (cov(Y33,Y37)>0) cov37_3=cov(Y33,Y37) else cov37_3=0
if (cov(Y34,Y35)>0) cov45_3=cov(Y34,Y35) else cov45_3=0
if (cov(Y34,Y36)>0) cov46_3=cov(Y34,Y36) else cov46_3=0
if (cov(Y34,Y37)>0) cov47_3=cov(Y34,Y37) else cov47_3=0
if (cov(Y35,Y36)>0) cov56_3=cov(Y35,Y36) else cov56_3=0
if (cov(Y35,Y37)>0) cov57_3=cov(Y35,Y37) else cov57_3=0
if (cov(Y36,Y37)>0) cov67_3=cov(Y36,Y37) else cov67_3=0

cov3=c(cov12_3,cov13_3,cov14_3,cov15_3,cov16_3,cov17_3,cov23_3,cov24_3,
      cov25_3,cov26_3,cov27_3,cov34_3,cov35_3,cov36_3,cov37_3,cov45_3,
      cov46_3,cov47_3,cov56_3,cov57_3,cov67_3)

theta38=min(mean(cov3),(alfa3/(1+exp(-1*(delta3+gama3*t[1]))))+beta1,
            (alfa3/(1+exp(-1*(delta3+gama3*t[1]))))+beta2,
            (alfa3/(1+exp(-1*(delta3+gama3*t[1]))))+beta3,
            (alfa3/(1+exp(-1*(delta3+gama3*t[1]))))+beta4,
            (alfa3/(1+exp(-1*(delta3+gama3*t[1]))))+beta5,
            (alfa3/(1+exp(-1*(delta3+gama3*t[2]))))+beta1,
            (alfa3/(1+exp(-1*(delta3+gama3*t[2]))))+beta2,
            (alfa3/(1+exp(-1*(delta3+gama3*t[2]))))+beta3,
            (alfa3/(1+exp(-1*(delta3+gama3*t[2]))))+beta4,
            (alfa3/(1+exp(-1*(delta3+gama3*t[2]))))+beta5,
            (alfa3/(1+exp(-1*(delta3+gama3*t[3]))))+beta1,
            (alfa3/(1+exp(-1*(delta3+gama3*t[3]))))+beta2,
            (alfa3/(1+exp(-1*(delta3+gama3*t[3]))))+beta3,
            (alfa3/(1+exp(-1*(delta3+gama3*t[3]))))+beta4,
            (alfa3/(1+exp(-1*(delta3+gama3*t[3]))))+beta5,
            (alfa3/(1+exp(-1*(delta3+gama3*t[4]))))+beta1,
            (alfa3/(1+exp(-1*(delta3+gama3*t[4]))))+beta2,
            (alfa3/(1+exp(-1*(delta3+gama3*t[4]))))+beta3,
            (alfa3/(1+exp(-1*(delta3+gama3*t[4]))))+beta4,
            (alfa3/(1+exp(-1*(delta3+gama3*t[4]))))+beta5,
            (alfa3/(1+exp(-1*(delta3+gama3*t[5]))))+beta1,
            (alfa3/(1+exp(-1*(delta3+gama3*t[5]))))+beta2,
            (alfa3/(1+exp(-1*(delta3+gama3*t[5]))))+beta3,

```



```

(alfa3/(1+exp(-1*(delta3+gama3*t[5]))))+beta4,
(alfa3/(1+exp(-1*(delta3+gama3*t[5]))))+beta5,
(alfa3/(1+exp(-1*(delta3+gama3*t[6]))))+beta1,
(alfa3/(1+exp(-1*(delta3+gama3*t[6]))))+beta2,
(alfa3/(1+exp(-1*(delta3+gama3*t[6]))))+beta3,
(alfa3/(1+exp(-1*(delta3+gama3*t[6]))))+beta4,
(alfa3/(1+exp(-1*(delta3+gama3*t[6]))))+beta5,
(alfa3/(1+exp(-1*(delta3+gama3*t[7]))))+beta1,
(alfa3/(1+exp(-1*(delta3+gama3*t[7]))))+beta2,
(alfa3/(1+exp(-1*(delta3+gama3*t[7]))))+beta3,
(alfa3/(1+exp(-1*(delta3+gama3*t[7]))))+beta4,
(alfa3/(1+exp(-1*(delta3+gama3*t[7]))))+beta5)
theta38_ini=theta38

if (cov(Y41,Y42)>0) cov12_4=cov(Y41,Y42) else cov12_4=0
if (cov(Y41,Y43)>0) cov13_4=cov(Y41,Y43) else cov13_4=0
if (cov(Y41,Y44)>0) cov14_4=cov(Y41,Y44) else cov14_4=0
if (cov(Y41,Y45)>0) cov15_4=cov(Y41,Y45) else cov15_4=0
if (cov(Y41,Y46)>0) cov16_4=cov(Y41,Y46) else cov16_4=0
if (cov(Y41,Y47)>0) cov17_4=cov(Y41,Y47) else cov17_4=0
if (cov(Y42,Y43)>0) cov23_4=cov(Y42,Y43) else cov23_4=0
if (cov(Y42,Y44)>0) cov24_4=cov(Y42,Y44) else cov24_4=0
if (cov(Y42,Y45)>0) cov25_4=cov(Y42,Y45) else cov25_4=0
if (cov(Y42,Y46)>0) cov26_4=cov(Y42,Y46) else cov26_4=0
if (cov(Y42,Y47)>0) cov27_4=cov(Y42,Y47) else cov27_4=0
if (cov(Y43,Y44)>0) cov34_4=cov(Y43,Y44) else cov34_4=0
if (cov(Y43,Y45)>0) cov35_4=cov(Y43,Y45) else cov35_4=0
if (cov(Y43,Y46)>0) cov36_4=cov(Y43,Y46) else cov36_4=0
if (cov(Y43,Y47)>0) cov37_4=cov(Y43,Y47) else cov37_4=0
if (cov(Y44,Y45)>0) cov45_4=cov(Y44,Y45) else cov45_4=0
if (cov(Y44,Y46)>0) cov46_4=cov(Y44,Y46) else cov46_4=0
if (cov(Y44,Y47)>0) cov47_4=cov(Y44,Y47) else cov47_4=0
if (cov(Y45,Y46)>0) cov56_4=cov(Y45,Y46) else cov56_4=0
if (cov(Y45,Y47)>0) cov57_4=cov(Y45,Y47) else cov57_4=0
if (cov(Y46,Y47)>0) cov67_4=cov(Y46,Y47) else cov67_4=0

cov4=c(cov12_4,cov13_4,cov14_4,cov15_4,cov16_4,cov17_4,cov23_4,cov24_4,
cov25_4,cov26_4,cov27_4,cov34_4,cov35_4,cov36_4,cov37_4,cov45_4,
cov46_4,cov47_4,cov56_4,cov57_4,cov67_4)

theta48=min(mean(cov4),(alfa4/(1+exp(-1*(delta4+gama4*t[1]))))+beta1,
(alfa4/(1+exp(-1*(delta4+gama4*t[1]))))+beta2,
(alfa4/(1+exp(-1*(delta4+gama4*t[1]))))+beta3,
(alfa4/(1+exp(-1*(delta4+gama4*t[1]))))+beta4,
(alfa4/(1+exp(-1*(delta4+gama4*t[1]))))+beta5,

```



```

(alfa4/(1+exp(-1*(delta4+gama4*t[2]))))+beta1,
(alfa4/(1+exp(-1*(delta4+gama4*t[2]))))+beta2,
(alfa4/(1+exp(-1*(delta4+gama4*t[2]))))+beta3,
(alfa4/(1+exp(-1*(delta4+gama4*t[2]))))+beta4,
(alfa4/(1+exp(-1*(delta4+gama4*t[2]))))+beta5,
(alfa4/(1+exp(-1*(delta4+gama4*t[3]))))+beta1,
(alfa4/(1+exp(-1*(delta4+gama4*t[3]))))+beta2,
(alfa4/(1+exp(-1*(delta4+gama4*t[3]))))+beta3,
(alfa4/(1+exp(-1*(delta4+gama4*t[3]))))+beta4,
(alfa4/(1+exp(-1*(delta4+gama4*t[3]))))+beta5,
(alfa4/(1+exp(-1*(delta4+gama4*t[4]))))+beta1,
(alfa4/(1+exp(-1*(delta4+gama4*t[4]))))+beta2,
(alfa4/(1+exp(-1*(delta4+gama4*t[4]))))+beta3,
(alfa4/(1+exp(-1*(delta4+gama4*t[4]))))+beta4,
(alfa4/(1+exp(-1*(delta4+gama4*t[4]))))+beta5,
(alfa4/(1+exp(-1*(delta4+gama4*t[5]))))+beta1,
(alfa4/(1+exp(-1*(delta4+gama4*t[5]))))+beta2,
(alfa4/(1+exp(-1*(delta4+gama4*t[5]))))+beta3,
(alfa4/(1+exp(-1*(delta4+gama4*t[5]))))+beta4,
(alfa4/(1+exp(-1*(delta4+gama4*t[5]))))+beta5,
(alfa4/(1+exp(-1*(delta4+gama4*t[6]))))+beta1,
(alfa4/(1+exp(-1*(delta4+gama4*t[6]))))+beta2,
(alfa4/(1+exp(-1*(delta4+gama4*t[6]))))+beta3,
(alfa4/(1+exp(-1*(delta4+gama4*t[6]))))+beta4,
(alfa4/(1+exp(-1*(delta4+gama4*t[6]))))+beta5,
(alfa4/(1+exp(-1*(delta4+gama4*t[7]))))+beta1,
(alfa4/(1+exp(-1*(delta4+gama4*t[7]))))+beta2,
(alfa4/(1+exp(-1*(delta4+gama4*t[7]))))+beta3,
(alfa4/(1+exp(-1*(delta4+gama4*t[7]))))+beta4,
(alfa4/(1+exp(-1*(delta4+gama4*t[7]))))+beta5)
theta48_ini=theta48

psi1=(var(Y11)-((alfa1/(1+exp(-1*(delta1+gama1*t[1]))))+beta1))/
      ((alfa1/(1+exp(-1*(delta1+gama1*t[1]))))+beta1)^2

if (psi1>0) psi=psi1 else psi=0.01
psi_ini=psi

p=c(alfa1_ini,delta1_ini,gama1_ini,theta18_ini,
    alfa2_ini,delta2_ini,gama2_ini,theta28_ini,
    alfa3_ini,delta3_ini,gama3_ini,theta38_ini,
    alfa4_ini,delta4_ini,gama4_ini,theta48_ini,
    beta2_ini,beta3_ini,beta4_ini,beta5_ini,psi_ini)

#####Início do Processo Iterativo

```

```

nk=length(Yt1)
m2lnL=1000; m2lnlike=1000
m2lnL1=0; m2lnL2=0
iter=0
difllike1=1000; difllike2=1000
prec=1e-6; maxit=500

while ((difllike1 > prec) & (iter < maxit )) {
  iter=iter+1
  m2lnL1=m2lnL; m2lnL2=m2lnlike

  #####PASSO E#####/
  ##### Esperança condicional de Zij/Yij
  Z=zesper(p)

  ##### Esperança condicional de (Wij8/Zij,Yij)
  ew8=fw8dyz(p)

  #####PASSO M#####/
  ##### Maximização da função de verossimilhança
  out1<-optim(c(alfa1,delta1,gama1,theta18,alfa2,delta2,gama2,theta28,
               alfa3,delta3,gama3,theta38,alfa4,delta4,gama4,theta48,
               beta2,beta3,beta4,beta5,psi),mlnv,sgrd,method="BFGS",
               control=list(maxit=500),hessian=TRUE)

  alfa1=out1$par[1];delta1=out1$par[2];gama1=out1$par[3];theta18=out1$par[4]
  alfa2=out1$par[5];delta2=out1$par[6];gama2=out1$par[7];theta28=out1$par[8]
  alfa3=out1$par[9];delta3=out1$par[10];gama3=out1$par[11];theta38=out1$par[12]
  alfa4=out1$par[13];delta4=out1$par[14];gama4=out1$par[15];theta48=out1$par[16]

  beta1=0;
  beta2=out1$par[17];beta3=out1$par[18];beta4=out1$par[19];beta5=out1$par[20]

  psi=out1$par[21]

  p=c(alfa1,delta1,gama1,theta18,alfa2,delta2,gama2,theta28,
      alfa3,delta3,gama3,theta38,alfa4,delta4,gama4,theta48,
      beta2,beta3,beta4,beta5,psi)

  converge=out1$convergence; names(converge)<-('converge')

  m2lnL=2*out1$value; m2lnlike=-2*lvyp(p)
  difllike1=abs((m2lnL-m2lnL1)/m2lnL1); difllike2=abs((m2lnlike-m2lnL2)/m2lnL2)

  names(nsimul) <- ('Simul'); names(iter)<-('iter')

```

```

printvectort1<-c(alfa1,delta1,gama1,theta18)
printvectort2<-c(alfa2,delta2,gama2,theta28)
printvectort3<-c(alfa3,delta3,gama3,theta38)
printvectort4<-c(alfa4,delta4,gama4,theta48)
printvectorbeta<-c(beta1,beta2,beta3,beta4,beta5)
printvectorpsi<-c(psi)
printvectorvero<-c(m2lnL,difllike1,m2lnlike,difllike2)

names(printvectort1)<-c('alfa1','delta1','gama1','theta18')
names(printvectort2)<-c('alfa2','delta2','gama2','theta28')
names(printvectort3)<-c('alfa3','delta3','gama3','theta38')
names(printvectort4)<-c('alfa4','delta4','gama4','theta48')
names(printvectorbeta)<-c('beta1','beta2','beta3','beta4','beta5')
names(printvectorpsi)<-c('psi')
names(printvectorvero)<-c('-2*logvero1','Rel.Dif.logvero1',
                          '-2*logvero2','Rel.Dif.logvero2')

print.default(nsimul); print.default(iter)
print.default(converge,digits=10)
print.default(printvectort1,digits=10)
print.default(printvectort2,digits=10)
print.default(printvectort3,digits=10)
print.default(printvectort4,digits=10)
print.default(printvectorbeta,digists=10)
print.default(printvectorpsi,digits=10)
print.default(printvectorvero,digits=10)
}
result[k,]=c(nsimul,iter,converge,m2lnL,difllike1,m2lnlike,difllike2,
             alfa1_ini,delta1_ini,gama1_ini,theta18_ini,
             alfa2_ini,delta2_ini,gama2_ini,theta28_ini,
             alfa3_ini,delta3_ini,gama3_ini,theta38_ini,
             alfa4_ini,delta4_ini,gama4_ini,theta48_ini,
             beta1_ini,beta2_ini,beta3_ini,beta4_ini,beta5_ini,psi_ini)
result[k+1,]=c(nsimul,iter,converge,m2lnL,difllike1,m2lnlike,difllike2,
               alfa1,delta1,gama1,theta18,alfa2,delta2,gama2,theta28,
               alfa3,delta3,gama3,theta38,alfa4,delta4,gama4,theta48,
               beta1,beta2,beta3,beta4,beta5,psi)
k=k+2
nsimul=nsimul+1
}
)

```

ANEXOS

ANEXO A - Cálculo da matriz de covariâncias entre observações de um mesmo bloco - Modelo com efeito aleatório de blocos

A.1 - Covariâncias entre observações de um mesmo bloco e mesmo tratamento

$$Cov(\mathbf{Y}_{ij}, \mathbf{Y}_{ij}^t) = E[Cov(\mathbf{Y}_{ij}, \mathbf{Y}_{ij}^t) | b_i] + Cov[E(\mathbf{Y}_{ij} | b_i), E(\mathbf{Y}_{ij}^t | b_i)] \quad (23)$$

Mas

$$Cov(\mathbf{Y}_{ij}, \mathbf{Y}_{ij}^t | b_i) = \begin{bmatrix} f_j(tempo_1, \beta_j) + b_i & \theta_{j,T+1} & \dots & \theta_{j,T+1} \\ & f_j(tempo_2, \beta_j) + b_i & \dots & \theta_{j,T+1} \\ & & \ddots & \vdots \\ & & & f_j(tempo_T, \beta_j) + b_i \end{bmatrix}.$$

Como $b_i \sim Gama(1/\sigma_b^2, 1/\sigma_b^2)$, com $E(b_i) = 1$ e $V(b_i) = \sigma_b^2$, então

$$E[Cov(\mathbf{Y}_{ij}, \mathbf{Y}_{ij}^t) | b_i] = \begin{bmatrix} f_j(tempo_1, \beta_j) + 1 & \theta_{j,T+1} & \dots & \theta_{j,T+1} \\ & f_j(tempo_2, \beta_j) + 1 & \dots & \theta_{j,T+1} \\ & & \ddots & \vdots \\ & & & f_j(tempo_T, \beta_j) + 1 \end{bmatrix}. \quad (24)$$

O segundo termo do lado direito da igualdade em (23) é dado por

$$Cov[E(\mathbf{Y}_{ij} | b_i), E(\mathbf{Y}_{ij}^t | b_i)] = E[E(\mathbf{Y}_{ij} | b_i)E(\mathbf{Y}_{ij}^t | b_i)] - E[E(\mathbf{Y}_{ij} | b_i)]E[E(\mathbf{Y}_{ij}^t | b_i)]. \quad (25)$$

Encontrando os termos da igualdade em (25), tem-se:

(i)

$$E[E(\mathbf{Y}_{ij} | b_i)E(\mathbf{Y}_{ij}^t | b_i)] = \begin{bmatrix} E\{[f_j(1, \beta_j) + b_i]^2\} & E\{[f_j(1, \beta_j) + b_i][f_j(2, \beta_j) + b_i]\} & \dots & E\{[f_j(1, \beta_j) + b_i][f_j(T, \beta_j) + b_i]\} \\ & E\{[f_j(2, \beta_j) + b_i]^2\} & \dots & E\{[f_j(2, \beta_j) + b_i][f_j(T, \beta_j) + b_i]\} \\ & & \ddots & \vdots \\ & & & E\{[f_j(T, \beta_j) + b_i]^2\} \end{bmatrix}$$

em que $f_j(t, \beta_j) = f_j(tempo_t, \beta_j)$, para todo $t = 1, 2, \dots, T$,

$$E\{[f_j(t, \beta_j) + b_i]^2\} = [f_j(t, \beta_j)]^2 + 2f_j(t, \beta_j) + \sigma_b^2 + 1 \quad (26)$$

$$E\{[f_j(t, \beta_j) + b_i][f_j(t', \beta_j) + b_i]\} = f_j(t, \beta_j)f_j(t', \beta_j) + f_j(t, \beta_j) + f_j(t', \beta_j) + \sigma_b^2 + 1, \quad (27)$$

para todo $t, t' = 1, 2, \dots, T$

(ii)

$$E[E(\mathbf{Y}_{ij} | b_i)] = \begin{bmatrix} E[f_j(1, \beta_j) + b_i] \\ E[f_j(2, \beta_j) + b_i] \\ \vdots \\ E[f_j(T, \beta_j) + b_i] \end{bmatrix} = \begin{bmatrix} f_j(1, \beta_j) + 1 \\ f_j(2, \beta_j) + 1 \\ \vdots \\ f_j(T, \beta_j) + 1 \end{bmatrix}. \quad (28)$$

Como $E[E(\mathbf{Y}_{ij}^t | b_i)] = E[E(\mathbf{Y}_{ij}^t | b_i)]^t$, obtém-se:

$$E[E(\mathbf{Y}_{ij} | b_i)]E[E(\mathbf{Y}_{ij}^t | b_i)] = \begin{bmatrix} [f_j(1, \boldsymbol{\beta}_j) + 1]^2 & [f_j(1, \boldsymbol{\beta}_j) + 1][f_j(2, \boldsymbol{\beta}_j) + 1] & \dots & [f_j(1, \boldsymbol{\beta}_j) + 1][f_j(T, \boldsymbol{\beta}_j) + 1] \\ & [f_j(2, \boldsymbol{\beta}_j) + 1]^2 & \dots & [f_j(2, \boldsymbol{\beta}_j) + 1][f_j(T, \boldsymbol{\beta}_j) + 1] \\ & & \ddots & \vdots \\ & & & [f_j(T, \boldsymbol{\beta}_j) + 1]^2 \end{bmatrix} \quad (29)$$

Pelas matrizes (26) e (29) e pelas expressões (26) e (27), obtém-se que

$$Cov[E(\mathbf{Y}_{ij} | b_i), E(\mathbf{Y}_{ij}^t | b_i)] = \begin{bmatrix} \sigma_b^2 & \sigma_b^2 & \dots & \sigma_b^2 \\ & \sigma_b^2 & \dots & \sigma_b^2 \\ & & \ddots & \vdots \\ & & & \sigma_b^2 \end{bmatrix}. \quad (30)$$

Por (24) e (30) chega-se ao resultado:

$$Cov(\mathbf{Y}_{ij}, \mathbf{Y}_{ij}^t) = \begin{bmatrix} f_j(1, \boldsymbol{\beta}_j) + \sigma_b^2 + 1 & \theta_{j,T+1} + \sigma_b^2 & \dots & \theta_{j,T+1} + \sigma_b^2 \\ & f_j(2, \boldsymbol{\beta}_j) + \sigma_b^2 + 1 & \dots & \theta_{j,T+1} + \sigma_b^2 \\ & & \ddots & \vdots \\ & & & f_j(T, \boldsymbol{\beta}_j) + \sigma_b^2 + 1 \end{bmatrix}.$$

A.2 - Covariâncias entre observações de um mesmo bloco e tratamentos diferentes

$$Cov(\mathbf{Y}_{ij}, \mathbf{Y}_{ij'}^t) = E[Cov(\mathbf{Y}_{ij}, \mathbf{Y}_{ij'}^t) | b_i] + Cov[E(\mathbf{Y}_{ij} | b_i), E(\mathbf{Y}_{ij'}^t | b_i)]. \quad (31)$$

Mas $\mathbf{Y}_{ij}$ e $\mathbf{Y}_{ij'}$ são condicionalmente independentes, o que resulta em $Cov[(\mathbf{Y}_{ij}, \mathbf{Y}_{ij'}^t) | b_i] = \mathbf{0}_{(T+1, T+1)}$, e assim, $Cov(\mathbf{Y}_{ij}, \mathbf{Y}_{ij'}^t) = Cov[E(\mathbf{Y}_{ij} | b_i), E(\mathbf{Y}_{ij'}^t | b_i)]$, sendo

$$Cov[E(\mathbf{Y}_{ij} | b_i), E(\mathbf{Y}_{ij'}^t | b_i)] = E[E(\mathbf{Y}_{ij} | b_i)E(\mathbf{Y}_{ij'}^t | b_i)] - E[E(\mathbf{Y}_{ij} | b_i)]E[E(\mathbf{Y}_{ij'}^t | b_i)].$$

Mas

$$E[E(\mathbf{Y}_{ij} | b_i)E(\mathbf{Y}_{ij'}^t | b_i)] = \begin{bmatrix} E\{[f_j(1, \boldsymbol{\beta}_j) + b_i][f_{j'}(1, \boldsymbol{\beta}_{j'}) + b_i]\} & E\{[f_j(1, \boldsymbol{\beta}_j) + b_i][f_{j'}(2, \boldsymbol{\beta}_{j'}) + b_i]\} & \dots & E\{[f_j(1, \boldsymbol{\beta}_j) + b_i][f_{j'}(T, \boldsymbol{\beta}_{j'}) + b_i]\} \\ E\{[f_j(2, \boldsymbol{\beta}_j) + b_i][f_{j'}(2, \boldsymbol{\beta}_{j'}) + b_i]\} & \dots & \dots & E\{[f_j(2, \boldsymbol{\beta}_j) + b_i][f_{j'}(T, \boldsymbol{\beta}_{j'}) + b_i]\} \\ & \ddots & \vdots & \\ & & E\{[f_j(T, \boldsymbol{\beta}_j) + b_i][f_{j'}(T, \boldsymbol{\beta}_{j'}) + b_i]\} \end{bmatrix}, \quad (32)$$

em que

$$E\{[f_j(t, \boldsymbol{\beta}_j) + b_i][f_{j'}(t', \boldsymbol{\beta}_{j'}) + b_i]\} = f_j(t, \boldsymbol{\beta}_j)f_{j'}(t', \boldsymbol{\beta}_{j'}) + f_j(t, \boldsymbol{\beta}_j) + f_{j'}(t', \boldsymbol{\beta}_{j'}) + \sigma_b^2 + 1 \quad (33)$$

para todo $t, t' = 1, 2, \dots, T$.

Além disso, $E[E(\mathbf{Y}_{ij} \mid b_i)]$ é dada pelo vetor em (28), e, portanto,

$$E[E(\mathbf{Y}_{ij} \mid b_i)]E[E(\mathbf{Y}_{ij'}^t \mid b_i)] = \begin{bmatrix} [f_j(1, \boldsymbol{\beta}_j) + 1][f_{j'}(1, \boldsymbol{\beta}_{j'}) + 1] & [f_j(1, \boldsymbol{\beta}_j) + 1][f_{j'}(2, \boldsymbol{\beta}_{j'}) + 1] & \dots & [f_j(1, \boldsymbol{\beta}_j) + 1][f_{j'}(T, \boldsymbol{\beta}_{j'}) + 1] \\ & [f_j(2, \boldsymbol{\beta}_j) + 1][f_{j'}(2, \boldsymbol{\beta}_{j'}) + 1] & \dots & [f_j(2, \boldsymbol{\beta}_j) + 1][f_{j'}(T, \boldsymbol{\beta}_{j'}) + 1] \\ & & \ddots & \vdots \\ & & & [f_j(T, \boldsymbol{\beta}_j) + 1][f_{j'}(T, \boldsymbol{\beta}_{j'}) + 1] \end{bmatrix}, \quad (34)$$

em que

$$[f_j(t, \boldsymbol{\beta}_j) + 1][f_{j'}(t', \boldsymbol{\beta}_{j'}) + 1] = f_j(t, \boldsymbol{\beta}_j)f_{j'}(t', \boldsymbol{\beta}_{j'}) + f_j(t, \boldsymbol{\beta}_j) + f_{j'}(t', \boldsymbol{\beta}_{j'}) + 1. \quad (35)$$

Então, por (32), (33), (34) e (35), chega-se a

$$Cov(\mathbf{Y}_{ij}, \mathbf{Y}_{ij'}^t) = \begin{bmatrix} \sigma_b^2 & \sigma_b^2 & \dots & \sigma_b^2 \\ & \sigma_b^2 & \dots & \sigma_b^2 \\ & & \ddots & \vdots \\ & & & \sigma_b^2 \end{bmatrix}_{(T+1, T+1)}.$$

ANEXO B - Cálculo das esperanças condicionais utilizadas no algoritmo EM para delineamento com efeito aleatórios de blocos

B.1 - Esperança condicional $b_i^* = E(b_i | \mathbf{Y}_i, \boldsymbol{\beta}^{(p)}, \boldsymbol{\theta}_{T+1}^{(p)}, \sigma_b^{2(p)})$

Seja $\mathbf{Y}_i = (\mathbf{Y}_{i1}, \mathbf{Y}_{i2}, \dots, \mathbf{Y}_{ik})^t$ o vetor de observações do bloco i . Considere $b_i \sim \text{Gama}(1/\sigma_b^2, 1/\sigma_b^2)$.

$$\begin{aligned} E(b_i | \mathbf{Y}_i, \boldsymbol{\beta}^{(p)}, \boldsymbol{\theta}_{T+1}^{(p)}, \sigma_b^{2(p)}) &= \int_0^\infty b_i f(b_i | \mathbf{Y}_i, \boldsymbol{\beta}^{(p)}, \boldsymbol{\theta}_{T+1}^{(p)}, \sigma_b^{2(p)}) db_i \\ &= \int_0^\infty b_i \frac{f(b_i, \mathbf{Y}_i | \boldsymbol{\beta}^{(p)}, \boldsymbol{\theta}_{T+1}^{(p)}, \sigma_b^{2(p)})}{f(\mathbf{Y}_i | \boldsymbol{\beta}^{(p)}, \boldsymbol{\theta}_{T+1}^{(p)}, \sigma_b^{2(p)})} db_i \end{aligned}$$

Mas

$$\begin{aligned} f(\mathbf{Y}_i | \boldsymbol{\beta}^{(p)}, \boldsymbol{\theta}_{T+1}^{(p)}, \sigma_b^{2(p)}) &= \int_0^\infty f(\mathbf{Y}_i, b_i | \boldsymbol{\beta}^{(p)}, \boldsymbol{\theta}_{T+1}^{(p)}, \sigma_b^{2(p)}) db_i \\ &= \int_0^\infty f(\mathbf{Y}_i | b_i, \boldsymbol{\beta}^{(p)}, \boldsymbol{\theta}_{T+1}^{(p)}, \sigma_b^{2(p)}) f(b_i) db_i, \end{aligned}$$

e não depende de b_i . Portanto,

$$E(b_i | \mathbf{Y}_i, \boldsymbol{\beta}^{(p)}, \boldsymbol{\theta}_{T+1}^{(p)}, \sigma_b^{2(p)}) = \frac{\int_0^\infty b_i f(\mathbf{Y}_i | b_i, \boldsymbol{\beta}^{(p)}, \boldsymbol{\theta}_{T+1}^{(p)}, \sigma_b^{2(p)}) f(b_i) db_i}{\int_0^\infty f(\mathbf{Y}_i | b_i, \boldsymbol{\beta}^{(p)}, \boldsymbol{\theta}_{T+1}^{(p)}, \sigma_b^{2(p)}) f(b_i) db_i}. \quad (36)$$

O numerador de (36) é dado por:

$$\begin{aligned} \int_0^\infty b_i f(\mathbf{Y}_i | b_i, \boldsymbol{\beta}^{(p)}, \boldsymbol{\theta}_{T+1}^{(p)}, \sigma_b^{2(p)}) f(b_i) db_i &= \int_0^\infty b_i f(\mathbf{Y}_{i1}, \mathbf{Y}_{i2}, \dots, \mathbf{Y}_{ik} | b_i, \boldsymbol{\beta}^{(p)}, \boldsymbol{\theta}_{T+1}^{(p)}, \sigma_b^{2(p)}) f(b_i) db_i \\ &= \int_0^\infty \left\{ b_i \prod_{j=1}^k \left[\sum_{w_{ij,T+1}=0}^{\min(\mathbf{Y}_{ij})} \left(P(W_{ij,T+1} = w_{ij,T+1} | b_i, \boldsymbol{\beta}^{(p)}, \boldsymbol{\theta}_{T+1}^{(p)}, \sigma_b^{2(p)}) \right) \times \right. \right. \\ &\quad \times \left. \prod_{t=1}^T \left(P(W_{ij,t} = y_{ij,t} - w_{ij,T+1} | b_i, \boldsymbol{\beta}^{(p)}, \boldsymbol{\theta}_{T+1}^{(p)}, \sigma_b^{2(p)}) \right) \right] f(b_i) \right\} db_i \\ &= \int_0^\infty \left\{ b_i \prod_{j=1}^k \left[\sum_{w_{ij,T+1}=0}^{\min(\mathbf{Y}_{ij})} \left(\frac{e^{\theta_{j,T+1}} \theta_{j,T+1}^{w_{ij,T+1}}}{w_{ij,T+1}!} \prod_{t=1}^T \left(\frac{e^{-\theta_{ij,t}} \theta_{ij,t}^{y_{ij,t} - w_{ij,T+1}}}{(y_{ij,t} - w_{ij,T+1})!} \right) \right) \right] \times \right. \\ &\quad \times \left. \frac{1}{\Gamma(1/\sigma_b^2)} \left(\frac{1}{\sigma_b^2} \right)^{\frac{1}{\sigma_b^2}} b_i^{\frac{1}{\sigma_b^2}-1} e^{-\frac{b_i}{\sigma_b^2}} \right\} db_i, \end{aligned} \quad (37)$$

em que $\theta_{ij,t}$ é definido em (10).

O denominador de (36) é igual a:

$$\begin{aligned}
\int_0^\infty f(\mathbf{Y}_i | b_i, \boldsymbol{\beta}^{(p)}, \boldsymbol{\theta}_{T+1}^{(p)}, \sigma_b^{2(p)}) f(b_i) db_i &= \int_0^\infty f(\mathbf{Y}_{i1}, \mathbf{Y}_{i2}, \dots, \mathbf{Y}_{ik} | b_i, \boldsymbol{\beta}^{(p)}, \boldsymbol{\theta}_{T+1}^{(p)}, \sigma_b^{2(p)}) f(b_i) db_i \\
&= \int_0^\infty \left\{ \prod_{j=1}^k \left[\sum_{w_{ij,T+1}=0}^{\min(\mathbf{Y}_{ij})} \left(P(W_{ij,T+1} = w_{ij,T+1} | b_i, \boldsymbol{\beta}^{(p)}, \boldsymbol{\theta}_{T+1}^{(p)}, \sigma_b^{2(p)}) \right) \times \right. \right. \\
&\quad \times \left. \prod_{t=1}^T \left(P(W_{ij,t} = y_{ij,t} - w_{ij,T+1} | b_i, \boldsymbol{\beta}^{(p)}, \boldsymbol{\theta}_{T+1}^{(p)}, \sigma_b^{2(p)}) \right) \right] f(b_i) \right\} db_i \\
&= \int_0^\infty \left\{ \prod_{j=1}^k \left[\sum_{w_{ij,T+1}=0}^{\min(\mathbf{Y}_{ij})} \left(\frac{e^{\theta_{j,T+1}} \theta_{j,T+1}^{w_{ij,T+1}}}{w_{ij,T+1}!} \prod_{t=1}^T \left(\frac{e^{-\theta_{ij,t}} \theta_{ij,t}^{y_{ij,t} - w_{ij,T+1}}}{(y_{ij,t} - w_{ij,T+1})!} \right) \right) \right] \times \right. \\
&\quad \times \left. \frac{1}{\Gamma(1/\sigma_b^2)} \left(\frac{1}{\sigma_b^2} \right)^{\frac{1}{\sigma_b^2}} b_i^{\frac{1}{\sigma_b^2}-1} e^{-\frac{b_i}{\sigma_b^2}} \right\} db_i. \tag{38}
\end{aligned}$$

As integrais (37) e (38) são aproximadas por fórmula de quadratura de Gauss, mais especificamente pela fórmula de Gauss-Laguerre, em que $\int_0^\infty e^{-x} f(x) dx \simeq \sum_{q=1}^Q A_q f(x_q)$, sendo Q o número de pontos de quadratura, x_q os pontos de quadratura e A_q os valores dos pesos associados a eles, que podem ser encontrados em Abramowitz (1972).

B.2 - Esperança condicional $w_{ij,T+1}^* = E(W_{ij,T+1} | b_i^*, \mathbf{Y}_i, \boldsymbol{\beta}^{(p)}, \boldsymbol{\theta}_{T+1}^{(p)})$

$$E(W_{ij,T+1} | b_i^*, \mathbf{Y}_i, \boldsymbol{\beta}^{(p)}, \boldsymbol{\theta}_{T+1}^{(p)}) =$$

$$\sum_{w_{ij,T+1}=0}^{\min(\mathbf{Y}_{ij})} w_{ij,T+1} \frac{P(W_{ij,T+1} = w_{ij,T+1}, \mathbf{Y}_{ij} = y_{ij} | b_i^*, \mathbf{Y}_i, \boldsymbol{\beta}^{(p)}, \boldsymbol{\theta}_{T+1}^{(p)})}{P(\mathbf{Y}_{ij} = y_{ij} | b_i^*, \mathbf{Y}_i, \boldsymbol{\beta}^{(p)}, \boldsymbol{\theta}_{T+1}^{(p)})} =$$

$$\sum_{w_{ij,T+1}=0}^{\min(\mathbf{Y}_{ij})} \left\{ \frac{w_{ij,T+1} \left[P(W_{ij,T+1} = w_{ij,T+1} | b_i^*, \boldsymbol{\beta}^{(p)}, \boldsymbol{\theta}_{T+1}^{(p)}) \prod_{t=1}^T \left(P(W_{ij,t} = y_{ij,t} - w_{ij,T+1} | b_i^*, \boldsymbol{\beta}^{(p)}, \boldsymbol{\theta}_{T+1}^{(p)}) \right) \right]}{\sum_{w_{ij,T+1}=0}^{\min(\mathbf{Y}_{ij})} \left\{ P(W_{ij,T+1} = w_{ij,T+1} | b_i^*, \boldsymbol{\beta}^{(p)}, \boldsymbol{\theta}_{T+1}^{(p)}) \prod_{t=1}^T \left[P(W_{ij,t} = y_{ij,t} - w_{ij,T+1} | b_i^*, \boldsymbol{\beta}^{(p)}, \boldsymbol{\theta}_{T+1}^{(p)}, \sigma_b^{2(p)}) \right] \right\}} \right\} =$$

$$\frac{\sum_{w_{ij,T+1}=0}^{\min(\mathbf{Y}_{ij})} w_{ij,T+1} \left[P(W_{ij,T+1} = w_{ij,T+1} | b_i^*, \boldsymbol{\beta}^{(p)}, \boldsymbol{\theta}_{T+1}^{(p)}) \prod_{t=1}^T \left(P(W_{ij,t} = y_{ij,t} - w_{ij,T+1} | b_i^*, \boldsymbol{\beta}^{(p)}, \boldsymbol{\theta}_{T+1}^{(p)}) \right) \right]}{\sum_{w_{ij,T+1}=0}^{\min(\mathbf{Y}_{ij})} \left\{ P(W_{ij,T+1} = w_{ij,T+1} | b_i^*, \boldsymbol{\beta}^{(p)}, \boldsymbol{\theta}_{T+1}^{(p)}) \prod_{t=1}^T \left[P(W_{ij,t} = y_{ij,t} - w_{ij,T+1} | b_i^*, \boldsymbol{\beta}^{(p)}, \boldsymbol{\theta}_{T+1}^{(p)}, \sigma_b^{2(p)}) \right] \right\}},$$

resultando em

$$w_{ij,T+1}^* = \frac{\sum_{w_{ij,T+1}=0}^{\min(\mathbf{Y}_{ij})} w_{ij,T+1} p_{ij}}{\sum_{w_{ij,T+1}=0}^{\min(\mathbf{Y}_{ij})} p_{ij}}$$

em que

$$p_{ij} = \sum_{w_{ij,T+1}=0}^{\min(\mathbf{Y}_{ij})} \left[\frac{e^{\theta_{j,T+1}} \theta_{j,T+1}^{w_{ij,T+1}}}{w_{ij,T+1}!} \prod_{t=1}^T \left(\frac{e^{-\theta_{ij,t}} \theta_{ij,t}^{y_{ij,t} - w_{ij,T+1}}}{(y_{ij,t} - w_{ij,T+1})!} \right) \right].$$

ANEXO C - Cálculo das esperanças condicionais utilizadas no algoritmo EM na modelagem de superdispersão

C.1 - Esperança condicional $z_{ij}^* = E(Z_{ij} \mid \mathbf{Y}_{ij}, \boldsymbol{\theta}_j^{*(p)}, \theta_{j,T+1}^{(p)}, \sigma_z^{2(p)})$

Considere Z_{ij} uma variável aleatória tal que $Z_{ij} \sim \text{Gama}(1/\sigma_z^2, \sigma_z^2)$.

$$\begin{aligned}
 z_{ij}^* &= E(Z_{ij} \mid \mathbf{Y}_{ij}, \boldsymbol{\theta}_j^{*(p)}, \theta_{j,T+1}^{(p)}, \sigma_z^{2(p)}) = \int_0^\infty z_{ij} f(z_{ij} \mid \mathbf{Y}_{ij}, \boldsymbol{\theta}_j^{*(p)}, \theta_{j,T+1}^{(p)}, \sigma_z^{2(p)}) dz_{ij} \\
 &= \int_0^\infty z_{ij} \frac{f(z_{ij}, \mathbf{y}_{ij} \mid \boldsymbol{\theta}_j^{*(p)}, \theta_{j,T+1}^{(p)}, \sigma_z^{2(p)})}{f(\mathbf{y}_{ij} \mid \boldsymbol{\theta}_j^{*(p)}, \theta_{j,T+1}^{(p)}, \sigma_z^{2(p)})} dz_{ij} = \int_0^\infty z_{ij} \frac{f(z_{ij}, \mathbf{y}_{ij} \mid \boldsymbol{\theta}_j^{*(p)}, \theta_{j,T+1}^{(p)}, \sigma_z^{2(p)})}{\int_0^\infty f(\mathbf{y}_{ij}, z_{ij} \mid \boldsymbol{\theta}_j^{*(p)}, \theta_{j,T+1}^{(p)}, \sigma_z^{2(p)}) dz_{ij}} dz_{ij} \\
 &= \frac{\int_0^\infty z_{ij} f(z_{ij}, \mathbf{y}_{ij} \mid \boldsymbol{\theta}_j^{*(p)}, \theta_{j,T+1}^{(p)}, \sigma_z^{2(p)}) dz_{ij}}{\int_0^\infty f(\mathbf{y}_{ij} \mid \boldsymbol{\theta}_j^{*(p)}, \theta_{j,T+1}^{(p)}, \sigma_z^{2(p)}) f(z_{ij}) dz_{ij}} \quad (39)
 \end{aligned}$$

em que $\boldsymbol{\theta}_j^{*(p)}$ é dado por (19) em delineamentos completamente casualizados ou (20) para modelos com efeitos fixos de blocos.

Resolvendo a integral no numerador de (39) tem-se:

$$\begin{aligned}
 &\int_0^\infty z_{ij} f(z_{ij}, \mathbf{y}_{ij} \mid \boldsymbol{\theta}_j^{*(p)}, \theta_{j,T+1}^{(p)}) dz_{ij} \\
 &= \int_0^\infty z_{ij} f(\mathbf{y}_{ij} \mid z_{ij}, \boldsymbol{\theta}_j^{*(p)}, \theta_{j,T+1}^{(p)}) f(z_{ij}) dz_{ij} \\
 &= \int_0^\infty z_{ij} f(y_{ij1}, y_{ij2}, \dots, y_{ijT} \mid z_{ij}, \boldsymbol{\theta}_j^{*(p)}, \theta_{j,T+1}^{(p)}) f(z_{ij}) dz_{ij} \\
 &= \int_0^\infty z_{ij} \sum_{w_{ij,T+1}=0}^{\min(\mathbf{y}_{ij})} [P(W_{ij,T+1} = w_{ij,T+1}, W_{ij1} = y_{ij1} + w_{ij,T+1}, W_{ij2} = y_{ij2} + w_{ij,T+1}, \dots, \\
 &\quad \dots, W_{ijT} = y_{ijT} + w_{ij,T+1} \mid z_{ij}, \boldsymbol{\theta}_j^{*(p)}, \theta_{j,T+1}^{(p)})] f(z_{ij}) dz_{ij} \\
 &= \int_0^\infty \left\{ z_{ij} \sum_{w_{ij,T+1}=0}^{\min(\mathbf{y}_{ij})} \left[P(W_{ij,T+1} \mid z_{ij}, \boldsymbol{\theta}_j^{*(p)}, \theta_{j,T+1}^{(p)}) \times \right. \right. \\
 &\quad \times \prod_{t=1}^T \left(P(W_{ijt} = y_{ijt} - w_{ij,T+1} \mid z_{ij}, \boldsymbol{\theta}_j^{*(p)}, \theta_{j,T+1}^{(p)}) \right) \left. \right] f(z_{ij}) \Big\} dz_{ij}
 \end{aligned}$$

$$\begin{aligned}
&= \int_0^\infty \left\{ z_{ij} \sum_{w_{ij,T+1}=0}^{\min(\mathbf{y}_{ij})} \left[\frac{e^{-\theta_{j,T+1} z_{ij}} (\theta_{j,T+1} z_{ij})^{w_{ij,T+1}}}{w_{ij,T+1}!} \times \right. \right. \\
&\quad \times \left. \prod_{t=1}^T \left(\frac{e^{-\theta_{jt}^* z_{ij}} (\theta_{jt}^* z_{ij})^{y_{ijt}-w_{ij,T+1}}}{(y_{ijt}-w_{ij,T+1})!} \right) \right] \frac{1}{\Gamma(1/\sigma_z^2)} \left(\frac{1}{\sigma_z^2} \right)^{\frac{1}{\sigma_z^2}} z_{ij}^{\frac{1}{\sigma_z^2}-1} e^{-\frac{z_{ij}}{\sigma_z^2}} \left. \right\} dz_{ij} \\
&= \sum_{w_{ij,T+1}=0}^{\min(\mathbf{y}_{ij})} \left\{ \frac{\theta_{j,T+1}^{w_{ij,T+1}}}{w_{ij,T+1}!} \prod_{t=1}^T \left(\frac{\theta_{jt}^{y_{ijt}-w_{ij,T+1}}}{(y_{ijt}-w_{ij,T+1})!} \right) \frac{1}{\Gamma(1/\sigma_z^2)} \left(\frac{1}{\sigma_z^2} \right)^{\frac{1}{\sigma_z^2}} \times \right. \\
&\quad \times \int_0^\infty \left[\frac{z_{ij}^{w_{ij,T+1} + \sum_{t=1}^T (y_{ijt}-w_{ij,T+1}) + 1 + \frac{1}{\sigma_z^2} - 1}}{z_{ij}} e^{-(\theta_{j,T+1} + \sum_{t=1}^T \theta_{jt} + \frac{1}{\sigma_z^2}) z_{ij}} \right] dz_{ij} \left. \right\} \\
&= \sum_{w_{ij,T+1}=0}^{\min(\mathbf{y}_{ij})} \left\{ \frac{\theta_{j,T+1}^{w_{ij,T+1}}}{w_{ij,T+1}!} \prod_{t=1}^T \left(\frac{\theta_{jt}^{y_{ijt}-w_{ij,T+1}}}{(y_{ijt}-w_{ij,T+1})!} \right) \frac{1}{\Gamma(1/\sigma_z^2)} \left(\frac{1}{\sigma_z^2} \right)^{\frac{1}{\sigma_z^2}} \times \right. \\
&\quad \times \frac{\Gamma[w_{ij,T+1} + \sum_{t=1}^T (y_{ijt}-w_{ij,T+1}) + \frac{1}{\sigma_z^2} + 1]}{(\theta_{j,T+1} + \sum_{t=1}^T \theta_{jt} + \frac{1}{\sigma_z^2})^{w_{ij,T+1} + \sum_{t=1}^T (y_{ijt}-w_{ij,T+1}) + \frac{1}{\sigma_z^2} + 1}} \times \\
&\quad \times \int_0^\infty \left[\frac{1}{\Gamma[w_{ij,T+1} + \sum_{t=1}^T (y_{ijt}-w_{ij,T+1}) + \frac{1}{\sigma_z^2} + 1]} \times \right. \\
&\quad \times (\theta_{j,T+1} + \sum_{t=1}^T \theta_{jt} + \frac{1}{\sigma_z^2})^{w_{ij,T+1} + \sum_{t=1}^T (y_{ijt}-w_{ij,T+1}) + \frac{1}{\sigma_z^2} + 1} \times \\
&\quad \times \left. \frac{z_{ij}^{w_{ij,T+1} + \sum_{t=1}^T (y_{ijt}-w_{ij,T+1}) + \frac{1}{\sigma_z^2} + 1 - 1}}{z_{ij}} e^{-(\theta_{j,T+1} + \sum_{t=1}^T \theta_{jt} + \frac{1}{\sigma_z^2}) z_{ij}} \right] dz_{ij} \left. \right\} \quad (40)
\end{aligned}$$

A integral em (40) corresponde à integral de uma função densidade de probabilidade de uma distribuição Gama, e, portanto, é igual a 1. Assim, a expressão para o numerador de (39) é igual a:

$$\begin{aligned}
&\int_0^\infty z_{ij} f(\mathbf{y}_{ij} \mid z_{ij}, \boldsymbol{\theta}_j^{*(p)}, \theta_{j,T+1}^{(p)}) f(z_{ij}) dz_{ij} = \\
&\sum_{w_{ij,T+1}=0}^{\min(\mathbf{y}_{ij})} \left\{ \frac{\theta_{j,T+1}^{w_{ij,T+1}}}{w_{ij,T+1}!} \prod_{t=1}^T \left(\frac{\theta_{jt}^{y_{ijt}-w_{ij,T+1}}}{(y_{ijt}-w_{ij,T+1})!} \right) \frac{1}{\Gamma(1/\sigma_z^2)} \left(\frac{1}{\sigma_z^2} \right)^{\frac{1}{\sigma_z^2}} \times \right. \\
&\quad \times \frac{\Gamma[w_{ij,T+1} + \sum_{t=1}^T (y_{ijt}-w_{ij,T+1}) + \frac{1}{\sigma_z^2} + 1]}{(\theta_{j,T+1} + \sum_{t=1}^T \theta_{jt} + \frac{1}{\sigma_z^2})^{w_{ij,T+1} + \sum_{t=1}^T (y_{ijt}-w_{ij,T+1}) + \frac{1}{\sigma_z^2} + 1}} \left. \right\} \quad (41)
\end{aligned}$$

Resolvendo a integral do denominador de (39) tem-se:

$$\begin{aligned}
& \int_0^\infty f(z_{ij}, \mathbf{y}_{ij} \mid \boldsymbol{\theta}_j^{*(p)}, \theta_{j,T+1}^{(p)}) dz_{ij} \\
= & \int_0^\infty f(\mathbf{y}_{ij} \mid z_{ij}, \boldsymbol{\theta}_j^{*(p)}, \theta_{j,T+1}^{(p)}) f(z_{ij}) dz_{ij} \\
= & \int_0^\infty f(y_{ij1}, y_{ij2}, \dots, y_{ijT} \mid z_{ij}, \boldsymbol{\theta}_j^{*(p)}, \theta_{j,T+1}^{(p)}) f(z_{ij}) dz_{ij} \\
= & \int_0^\infty \left\{ \sum_{w_{ij,T+1}=0}^{\min(\mathbf{y}_{ij})} \left[\frac{e^{-\theta_{j,T+1} z_{ij}} (\theta_{j,T+1} z_{ij})^{w_{ij,T+1}}}{w_{ij,T+1}!} \times \right. \right. \\
& \times \prod_{t=1}^T \left(\frac{e^{-\theta_{jt}^* z_{ij}} (\theta_{jt}^* z_{ij})^{y_{ijt}-w_{ij,T+1}}}{(y_{ijt}-w_{ij,T+1})!} \right) \left. \frac{1}{\Gamma(1/\sigma_z^2)} \left(\frac{1}{\sigma_z^2} \right)^{\frac{1}{\sigma_z^2}} z_{ij}^{\frac{1}{\sigma_z^2}-1} e^{-\frac{z_{ij}}{\sigma_z^2}} \right\} dz_{ij} \\
= & \sum_{w_{ij,T+1}=0}^{\min(\mathbf{y}_{ij})} \left\{ \frac{\theta_{j,T+1}^{w_{ij,T+1}}}{w_{ij,T+1}!} \prod_{t=1}^T \left(\frac{\theta_{jt}^{y_{ijt}-w_{ij,T+1}}}{(y_{ijt}-w_{ij,T+1})!} \right) \frac{1}{\Gamma(1/\sigma_z^2)} \left(\frac{1}{\sigma_z^2} \right)^{\frac{1}{\sigma_z^2}} \times \right. \\
& \times \int_0^\infty \left[\frac{w_{ij,T+1} + \sum_{t=1}^T (y_{ijt}-w_{ij,T+1}) + \frac{1}{\sigma_z^2} - 1}{z_{ij}} e^{-(\theta_{j,T+1} + \sum_{t=1}^T \theta_{jt} + \frac{1}{\sigma_z^2}) z_{ij}} \right] dz_{ij} \Big\} \\
= & \sum_{w_{ij,T+1}=0}^{\min(\mathbf{y}_{ij})} \left\{ \frac{\theta_{j,T+1}^{w_{ij,T+1}}}{w_{ij,T+1}!} \prod_{t=1}^T \left(\frac{\theta_{jt}^{y_{ijt}-w_{ij,T+1}}}{(y_{ijt}-w_{ij,T+1})!} \right) \frac{1}{\Gamma(1/\sigma_z^2)} \left(\frac{1}{\sigma_z^2} \right)^{\frac{1}{\sigma_z^2}} \times \right. \\
& \times \frac{\Gamma[w_{ij,T+1} + \sum_{t=1}^T (y_{ijt}-w_{ij,T+1}) + \frac{1}{\sigma_z^2}]}{(\theta_{j,T+1} + \sum_{t=1}^T \theta_{jt} + \frac{1}{\sigma_z^2})^{w_{ij,T+1} + \sum_{t=1}^T (y_{ijt}-w_{ij,T+1}) + \frac{1}{\sigma_z^2}}} \times \\
& \times \int_0^\infty \left[\frac{1}{\Gamma[w_{ij,T+1} + \sum_{t=1}^T (y_{ijt}-w_{ij,T+1}) + \frac{1}{\sigma_z^2}]} \times \right. \\
& \times (\theta_{j,T+1} + \sum_{t=1}^T \theta_{jt} + \frac{1}{\sigma_z^2})^{w_{ij,T+1} + \sum_{t=1}^T (y_{ijt}-w_{ij,T+1}) + \frac{1}{\sigma_z^2}} \times \\
& \times \left. \frac{w_{ij,T+1} + \sum_{t=1}^T (y_{ijt}-w_{ij,T+1}) + \frac{1}{\sigma_z^2} - 1}{z_{ij}} e^{-(\theta_{j,T+1} + \sum_{t=1}^T \theta_{jt} + \frac{1}{\sigma_z^2}) z_{ij}} \right] dz_{ij} \Big\} \quad (42)
\end{aligned}$$

A integral em (42) corresponde à integral de uma função densidade de probabilidade de uma distribuição Gama, e, portanto, é igual a 1. Assim, a expressão para o numerador de (39) é igual a:

$$\begin{aligned}
\int_0^\infty f(\mathbf{y}_{ij} \mid z_{ij}, \boldsymbol{\theta}_j^{*(p)}, \theta_{j,T+1}^{(p)}, \sigma_z^2) f(z_{ij}) dz_{ij} &= \sum_{w_{ij,T+1}=0}^{\min(\mathbf{y}_{ij})} \frac{\theta_{j,T+1}^{w_{ij,T+1}}}{w_{ij,T+1}!} \prod_{t=1}^T \left(\frac{\theta_{jt}^{y_{ijt}-w_{ij,T+1}}}{(y_{ijt}-w_{ij,T+1})!} \right) \frac{1}{\Gamma(1/\sigma_z^2)} \left(\frac{1}{\sigma_z^2} \right)^{\frac{1}{\sigma_z^2}} \times \\
&\times \frac{\Gamma[w_{ij,T+1} + \sum_{t=1}^T (y_{ijt}-w_{ij,T+1}) + \frac{1}{\sigma_z^2}]}{(\theta_{j,T+1} + \sum_{t=1}^T \theta_{jt} + \frac{1}{\sigma_z^2})^{w_{ij,T+1} + \sum_{t=1}^T (y_{ijt}-w_{ij,T+1}) + \frac{1}{\sigma_z^2}}}.
\end{aligned}$$

C.2 - Esperança condicional $w_{ij,T+1}^* = E(W_{ij,T+1} \mid \mathbf{Y}_{ij}, z_{ij}^*, \boldsymbol{\theta}_j^{*(p)}, \theta_{j,T+1}^{(p)}, \sigma_z^{2(p)})$

$$\begin{aligned}
w_{ij,T+1}^* &= \sum_{w_{ij,T+1}=0}^{\min(\mathbf{y}_{ij})} w_{ij,T+1} P(W_{ij,T+1} = w_{ij,T+1} \mid \mathbf{Y}_{ij}, z_{ij}^*, \boldsymbol{\theta}_j^{*(p)}, \theta_{j,T+1}^{(p)}, \sigma_z^{2(p)}) \\
&= \sum_{w_{ij,T+1}=0}^{\min(\mathbf{y}_{ij})} w_{ij,T+1} \frac{P(W_{ij,T+1} = w_{ij,T+1}, \mathbf{Y}_{ij} = \mathbf{y}_{ij} \mid z_{ij}^*, \boldsymbol{\theta}_j^{*(p)}, \theta_{j,T+1}^{(p)}, \sigma_z^{2(p)})}{P(\mathbf{Y}_{ij} = \mathbf{y}_{ij} \mid z_{ij}^*, \boldsymbol{\theta}_j^{*(p)}, \theta_{j,T+1}^{(p)}, \sigma_z^{2(p)})} \\
&= \sum_{w_{ij,T+1}=0}^{\min(\mathbf{y}_{ij})} w_{ij,T+1} \frac{P(W_{ij,T+1} = w_{ij,T+1}, Y_{ij1} = y_{ij1}, Y_{ij2} = y_{ij2}, \dots, Y_{ijT} = y_{ijT} \mid z_{ij}^*, \boldsymbol{\theta}_j^{*(p)}, \theta_{j,T+1}^{(p)}, \sigma_z^{2(p)})}{P(Y_{ij1} = y_{ij1}, Y_{ij2} = y_{ij2}, \dots, Y_{ijT} = y_{ijT} \mid z_{ij}^*, \boldsymbol{\theta}_j^{*(p)}, \theta_{j,T+1}^{(p)}, \sigma_z^{2(p)})} \\
&= \sum_{w_{ij,T+1}=0}^{\min(\mathbf{y}_{ij})} w_{ij,T+1} \frac{p_{ijt}}{\sum_{w_{ij,T+1}=0}^{\min(\mathbf{y}_{ij})} p_{ijt}} \\
&= \frac{\sum_{w_{ij,T+1}=0}^{\min(\mathbf{y}_{ij})} w_{ij,T+1} p_{ijt}}{\sum_{w_{ij,T+1}=0}^{\min(\mathbf{y}_{ij})} p_{ijt}}
\end{aligned}$$

em que

$$\begin{aligned}
p_{ijt} &= P(W_{ij,T+1} = w_{ij,T+1} \mid z_{ij}^*, \boldsymbol{\theta}_j^{*(p)}, \theta_{j,T+1}^{(p)}, \sigma_z^{2(p)}) \prod_{t=1}^T P(W_{ijt} = y_{ijt} - w_{ij,T+1} \mid z_{ij}^*, \boldsymbol{\theta}_j^{*(p)}, \theta_{j,T+1}^{(p)}, \sigma_z^{2(p)}) \\
&= \frac{e^{-\theta_{j,T+1} z_{ij}} (\theta_{j,T+1} z_{ij}^*)^{w_{ij,T+1}}}{w_{ij,T+1}!} \prod_{t=1}^T \frac{e^{-\theta_{jt}^* z_{ij}} (\theta_{jt}^* z_{ij}^*)^{y_{ijt} - w_{ij,T+1}}}{(y_{ijt} - w_{ij,T+1})!}.
\end{aligned}$$

ANEXO D - Histogramas das distribuições das estimativas dos parâmetros

D.1 - Delineamento completamente aleatorizado

Figura 5 - Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 1 - Delineamento completamente aleatorizado

Figura 6 - Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 2 - Delineamento completamente aleatorizado

Figura 7 - Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 3 - Delineamento completamente aleatorizado

Figura 8 - Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 4 - Delineamento completamente aleatorizado

D.2 - Delineamento em blocos casualizados com efeitos fixos de bloco

Figura 9 - Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 1 - Delineamento em blocos casualizados com efeitos fixos de bloco

Figura 10 - Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 2 - Delineamento em blocos casualizados com efeitos fixos de bloco

Figura 11 - Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 3 - Delineamento em blocos casualizados com efeitos fixos de bloco

Figura 12 - Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 4 - Delineamento em blocos casualizados com efeitos fixos de bloco

Figura 13 - Histograma representativo das estimativas dos efeitos fixos de blocos

D.3 - Delineamento em blocos casualizados com efeitos aleatórios de bloco

Figura 14 - Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 1 - Delineamento em blocos casualizados com efeitos aleatórios de bloco

Figura 15 - Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 2 - Delineamento em blocos casualizados com efeitos aleatórios de bloco

Figura 16 - Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 3 - Delineamento em blocos casualizados com efeitos aleatórios de bloco

Figura 17 - Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 4 - Delineamento em blocos casualizados com efeitos aleatórios de bloco

Figura 18 - Histograma representativo das estimativas do parâmetro σ_b^2 - Delineamento em blocos casualizados com efeitos aleatórios de bloco

D.4 - Delineamento completamente aleatorizado na presença de superdispersão

Figura 19 - Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 1 - Delineamento completamente aleatorizado com superdispersão

Figura 20 - Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 2 - Delineamento completamente aleatorizado com superdispersão

Figura 21 - Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 3 - Delineamento completamente aleatorizado com superdispersão

Figura 22 - Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 4 - Delineamento completamente aleatorizado com superdispersão

Figura 23 - Histograma representativo das estimativas do parâmetro σ_z^2 - Delineamento completamente aleatorizado com superdispersão

D.5 - Delineamento em blocos casualizados com efeitos fixos de bloco, na presença de superdispersão

Figura 24 - Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 1 - Delineamento em blocos casualizados com efeitos fixos de bloco, com superdispersão

Figura 25 - Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 2 - Delineamento em blocos casualizados com efeitos fixos de bloco, com superdispersão

Figura 26 - Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 3 - Delineamento em blocos casualizados com efeitos fixos de bloco, com superdispersão

Figura 27 - Histograma representativo das estimativas para os parâmetros da curva logística e de covariância do Tratamento 4 - Delineamento em blocos casualizados com efeitos fixos de bloco, com superdispersão

Figura 28 - Histograma representativo das estimativas dos efeitos de blocos - Delineamento em blocos casualizados com efeitos fixos de bloco, com superdispersão

Figura 29 - Histograma representativo das estimativas do parâmetro σ_z^2 - Delineamento em blocos casualizados com efeitos fixos de bloco, com superdispersão

ANEXO E - Matrizes de variâncias e covariâncias observadas e correlações entre o número de folhas em bromélia em cada tratamento

Os valores nas diagonais representam as variâncias. Acima delas, encontram-se as covariâncias e abaixo, as correlações.

Tabela 18 - Covariâncias (parte superior) e correlações (parte inferior) entre número de folhas de cada tempo, para o tratamento 1

	Tempo(dias após o plantio)					
Tempo	5	173	229	285	341	435
5	9,67	-0,67	7,00	8,00	2,67	8,17
173	-0,13	2,67	2,67	4,67	8,00	8,67
229	0,72	0,52	10,00	11,00	12,67	15,00
285	0,61	0,67	0,82	18,00	17,67	27,00
341	0,16	0,94	0,77	0,80	27,00	29,50
435	0,40	0,81	0,72	0,97	0,87	42,92

Tabela 19 - Covariâncias (parte superior) e correlações (parte inferior) entre número de folhas de cada tempo, para o tratamento 2

	Tempo(dias após o plantio)					
Tempo	5	173	229	285	341	435
5	7,58	5,67	11,33	2,33	3,92	5,75
173	0,59	12,00	10,00	-7,33	1,00	-3,00
229	0,88	0,62	22,00	7,33	11,00	10,00
285	0,20	-0,50	0,369	18,00	10,00	13,67
341	0,48	0,10	0,79	0,80	8,92	8,08
435	0,58	-0,24	0,59	0,90	0,75	12,92

Tabela 20 - Covariâncias (parte superior) e correlações (parte inferior) entre número de folhas de cada tempo, para o tratamento 3

Tempo	Tempo(dias após o plantio)					
	5	173	229	285	341	435
5	3,00	4,17	6,83	10,17	7,50	7,00
173	0,78	9,58	10,42	12,42	10,25	8,17
229	0,93	0,79	8,25	18,58	12,75	12,17
285	0,91	0,62	0,70	42,25	32,75	30,17
341	0,85	0,65	0,58	0,98	26,25	23,50
435	0,87	0,57	0,62	0,99	0,99	21,67

Tabela 21 - Covariâncias (parte superior) e correlações (parte inferior) entre número de folhas de cada tempo, para o tratamento 4

Tempo	Tempo(dias após o plantio)					
	5	173	229	285	341	435
5	4,92	3,83	10,42	1,75	-3,25	3,50
173	0,49	12,33	6,50	6,50	15,17	12,67
229	0,98	0,39	22,92	3,92	-8,75	8,50
285	0,33	0,78	0,35	5,58	10,92	12,17
341	-0,24	0,70	-0,30	0,75	38,25	22,50
435	0,30	0,69	0,34	0,99	0,70	27,00

Tabela 22 - Covariâncias (parte superior) e correlações (parte inferior) entre número de folhas de cada tempo, para o tratamento 5

Tempo	Tempo(dias após o plantio)					
	5	173	229	285	341	435
5	1,58	-6,33	-4,42	0,08	-4,00	-2,33
173	-0,78	42,00	41,00	-0,33	14,33	12,67
229	-0,52	0,93	46,25	-1,92	3,67	6,33
285	0,04	-0,03	-0,19	2,25	6,67	6,33
341	-0,57	0,40	0,10	0,79	31,33	25,33
435	-0,39	0,41	0,20	0,89	0,95	22,67